NPSEC-93-005

NAVAL POSTGRADUATE SCHOOL

Monterey, California

New Algorithms for the Detection and Elimination of Sine Waves and Other Narrow-Band Signals in the Presence of Broadband Signals and Noise

by

Michael A. Soderstrand
University of California, Davis

K. V. Rangarao and H. H. Loomis
Naval Postgraduate School

October 1992

Naval Postgraduate School
Monterey, California 93943-5000

Rear Admiral R.W. West, Jr.
Superintendent

H. Schull
Provost

This work reported herein was partially supported by the Secretary of the Air Force.

This report was prepared by Michael Soderstrand, K. V. Rangarao, and H. H. Loomis, Jr.

Submitted by:

H. H. LOOMIS, JR.
Professor of Electrical and
Computer Engineering

Reviewed by:

MICHAEL MORGAN
Chairman
Department of Electrical and
Computer Engineering

Reviewed by:

PAUL J. MARTO
Dean of Research

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a REPORT SECURITY CLASSIFICATION Unclassified	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Distribution unlimited.
2b DECLASSIFICATION/DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S) NPSEC-93-005	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b OFFICE SYMBOL (If applicable) Code EC	7a NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000	7b ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION Secretary of the Air Force 8b OFFICE SYMBOL (If applicable) ACBMB	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code) The Pentagon, Room 5C132 Washington, D.C. 20330-1000	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
NEW ALGORITHMS FOR THE DETECTION AND ELIMINATION OF SINE WAVES AND OTHER NARROW-BAND SIGNALS IN THE PRESENCE OF BROADBAND SIGNALS AND NOISE

12 PERSONAL AUTHOR(S)
Michael A. Soderstrand, K. V. Rangarao, and H. H. Loomis

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Technical Report	FROM _____ TO _____	1992 October 1	79

16 SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	narrow-band interference removal adaptive interference removal pipeline digital filters

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

Four different classes of adaptive signal cancelers can be used to eliminate narrow-band interference from a broadband signal: (1) cascaded second-order notch filters, (2) high-order in-line notch filters, (3) second-order bandpass noise cancelers, and (4) high-order bandpass noise cancelers. Of the four, a structure based on second-order bandpass filters used as signal cancelers is found to perform better than the other structures. The adaptive algorithm for these filters has been proposed by Kwan and Martin and modified by Petraglia, Mitra, and Szczupak.

The Kwan and Martin structure can be reduced in hardware complexity without degrading performance using a new adaptive algorithm that out-performs any of the other known structures or algorithms. This new structure is particularly suited to the elimination of narrow-band interference in broadband Bi-Phase Shift-Key (BPSK) signals with and without background noise.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT	21 ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	Unclassified
22a NAME OF RESPONSIBLE INDIVIDUAL Herschel H. Loomis, Jr.	22b TELEPHONE (Include Area Code) (408) 656-3214 / 22c OFFICE SYMBOL EC/Lm

DD Form 1473, JUN 86 Previous editions are obsolete SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603

NEW ALGORITHMS FOR THE DETECTION AND ELIMINATION OF SINE WAVES AND OTHER NARROW-BAND SIGNALS IN THE PRESENCE OF BROADBAND SIGNALS AND NOISE[1]

Naval Postgraduate School Technical Report

Michael A. Soderstrand
Electrical Engineering and Computer Science Department
University of Calif., Davis, CA 95616

K.V. Rangarao and H.H. Loomis
Electrical and Computer Engineering Department
Naval Postgraduate School
Monterey, CA 93943-5100

ABSTRACT

Four different classes of adaptive signal cancelers can be used to eliminate narrow-band interference from a broadband signal: (1) cascaded second-order notch filters, (2) high-order in-line notch filters, (3) second-order bandpass noise cancelers, and (4) high-order bandpass noise cancelers. Of the four, a structure based on second-order bandpass filters used as signal cancelers is found to perform better than the other structures. The adaptive algorithm for these filters has been proposed by Kwan and Martin and modified by Petraglia, Mitra and Szczupak.

The Kwan and Martin structure can be reduced in hardware complexity without degrading performance using a new adaptive algorithm that out-performs any of the other known structures or algorithms. This new structure is particularly suited to the elimination of narrow-band interference in broadband Bi-Phase Shift-Key (BPSK) signals with and without background noise.

---

[1]This work was supported in part by the United States Air Force.

0.0 Introduction

A common signal processing problem is the reception of a relatively weak broad-band signal such as a spread-spectrum Bi-Phase Shift-Key (BPSK) modulated signal in the presence of narrow-band interference. These narrow-band interferers may have less energy than the broad-band signal, but because it is concentrated over a narrow bandwidth they mask out the broad-band signal. In the specific work that we are interested in, the goal is to eliminate only the narrow-band interference. Later processing will separate broad-band noise from the desired signal. However, this later processing will not work properly unless all narrow-band interference is eliminated before the broad-band processing is begun.

The most common approach to eliminating the narrow-band interference is to make use of a cascade of notch filters [1-8]. Based on extensive analysis and experimental work, Kwan and Martin developed a particularly nice structure [5] which was modified by Petraglia, Mitra, and others [7,8] and applied with good success. However, the hardware required to achieve this performance with more than two or three interfering signals becomes too complex for easy implementation, particularly when high sampling rate is required. One of the major contributions of this report is to introduce a modified Kwan-Martin algorithm that maintains the performance of the original algorithm but requires substantially less hardware for elimination of narrow-band interference when there are multiple interferers.

In this document we compare several approaches to elimination of narrow-band interference from broad-band signals. Sections 1 describes four different structures for adaptive interference cancellation. Section 2 compares the four sections and gives the basis for the selection of the second-order band-pass canceler as the best design. Section 3 gives detailed information on the new algorithm for the second-order band-pass canceler and extensive experimental data to demonstrate its advantages over other algorithms including the algorithm by Kwan and Martin. Section 4 gives detailed examples of the new algorithm's performance in cancelling narrow-band BPSK signals in the presence of broad-band BPSK signals and noise. Conclusions are given in Section 5 including a discussion of the hardware implementation of these filters. Section 6 contains the references for the report.

1.0 Adaptive Notch Filters

Notch filters for removing multiple narrow-band interference can be categorized into four broad categories illustrated in Figure 1-1. The first two categories, Figures 1a and 1b, are cascaded second order notches with each second-order section removing one frequency. The next two categories, Figures 1c and 1d, are higher-order notches that eliminate multiple frequencies. In all of the categories, it is possible to use FIR filters (ie: all zero filters) which are easily pipelined and can be made truly linear phase. However, IIR filters out perform FIR filters. Thus IIR pipelining may become an important issue.

1.1 Second-Order Cascaded Notch Filters

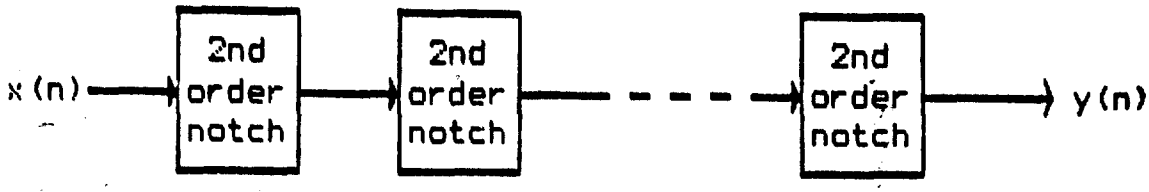
The second-order notch filter is used in cascade and in-line with the signal as shown in Figure 1-1a. The transfer function for the notch filter is given by:

$$H_N(z) = \frac{2-k_2}{2} \; \frac{1 - \dfrac{2(2-k_2-k_1^2)}{2-k_2} z^{-1} + z^{-2}}{1 - (2-k_2-k_1^2)z^{-1} + (1-k_2)z^{-2}} \qquad (1\text{-}1)$$

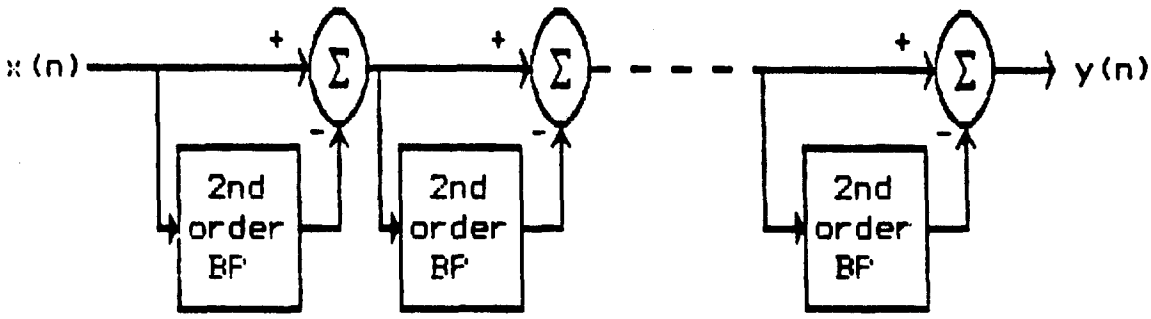
For arbitrary values of k_1 and k_2, this is a symmetric notch filter with unity gain at DC and the Nyquist frequency. If k_2 is kept constant, then the 3db notch width is also kept constant. Thus k_1 may be adapted to remove one narrow-band signal. A cascade of such filters can be used to remove multiple narrow-band signals.

1.2 Second-Order Cascaded Signal Canceler

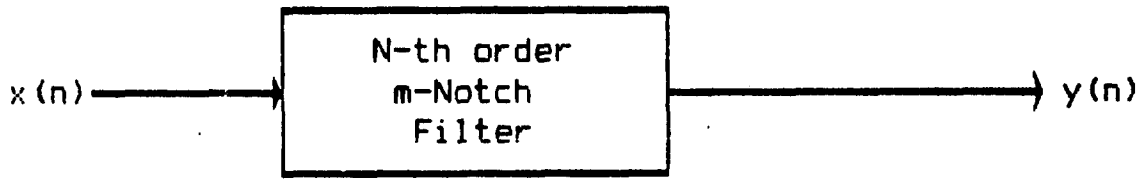
The cascaded second-order signal canceler approach shown in Figure 1-1b has the advantage that the desired signal does not pass through the adaptive filter. Instead, the band-pass filter is used to detect the narrow-band signal which is then subtracted from the desired signal. A constant 3db bandwidth notch can be achieved by selecting a band-pass filter with the transfer function:



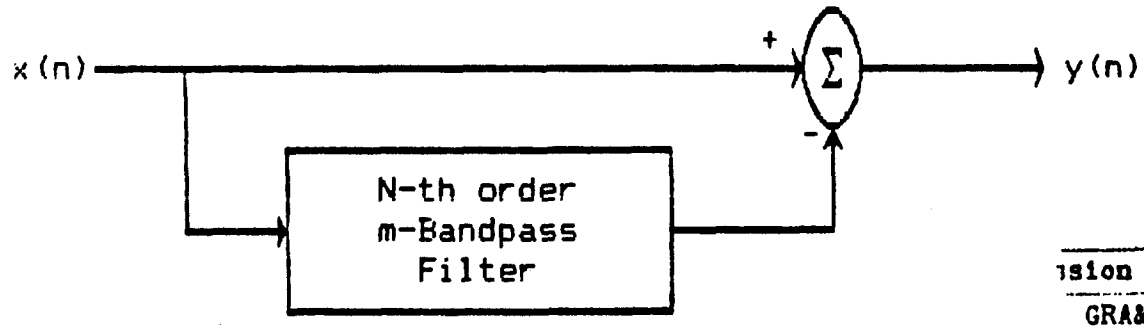
(a) Cascade of Second-Order Notch Filters



(b) Cascade of Second-Order Cancellation Filters



(c) Higher-Order Notch Filter



(d) Higher-Order Signal Cancelling Filter

Figure 1-1. Four Categories of Notch Filters

$$H_{BP}(z) = \frac{k_2(1 - z^{-2})}{2D(z)} \qquad (1\text{-}2)$$

where: $D(z)$ is the same denominator as $H_N(z)$ in equation (1).

The signal-canceler structure is also nice for adaptation because it is relatively easy to generate sensitivity functions which are related to the gradient of H_{BP} with respect to the frequency parameter k_1 [5]. Figure 1-2 shows the block diagram of an adaptive version of this filter. The sensitivity function $H_S(z)=H_{BP}(z)*H_{ss}(z)$ where:

$$H_{ss}(z) = \frac{2k_1 z^{-1}}{D(z)} \qquad (1\text{-}3)$$

The parameter k_1 may then be adapted by the formula:

$$k_1(n+1) = k_1(n) - \mu e(n)s(n) \qquad (3)$$

Considerable information is available in the literature on additional second-order notch filters [1-8]. At this time, however, the two discussed here seem to be best suited for our application.

1.3 Higher-Order In-Line Notch Filter

The notch filter of Figure 1-1c has the advantage that it is more easily pipelined. Such a filter could be constructed as a linear-phase FIR filter which is easy to pipeline and has the advantage of linear phase. However, an FIR filter will require many weights to obtain good performance.

An IIR in-line filter will allow good performance with much fewer weights. However, the IIR filter would be difficult to make adaptive. Pipelining the IIR filter is more difficult than

pipelining the FIR filter, but less difficult than pipelining the second-order IIR filter of the last section. This filter could be adapted by having a discrete set of pre-selected notches which could be switched in by a detection circuit that simply looks for maximums in the spectrum of the input signal. Although this approach could be used with any of the filters, it is particularly attractive with this filter due to the ability to generate geometric pole/zero patterns in higher-order filters.

1.4 Higher-Order Signal Canceler

The signal canceler of Figure 1-1d has most of the same benefits and disadvantages of the higher-order in-line notch discussed above. However, since the canceler is not in-line with the signal, the desired signal does not need to pass through the filter. Also, it may be easier to design the adaptive portion of this filter. An additional advantage comes from the ability to adapt the filter off-line and then switch it in once the filter is able to enhance the performance of the overall system.

1.5 IIR vs FIR Realizations

In our case of narrow-band interference, the second order notch of section 1.1 and the second-order band-pass canceler of section 1.2 would by necessity have to be IIR realizations. A second-order FIR filter cannot achieve a sufficiently narrow band-width in order to adequately handle the interfering signals. Theoretically, the higher-order realizations of sections 1.3 and 1.4 could be either IIR or FIR. However, the second-order IIR is ideally suited for our situation since it provides a sharp resonance with only one parameter, the notch or band-pass frequency, to be adapted. Higher order IIR filters would add great complication without significantly improving performance.

In many applications, however, it may be advantageous to use FIR rather than IIR filters. FIR filters must be higher order in order to realize the narrow-band characteristics. The two primary advantages of the FIR realizations are that they can be made to be linear phase and that they can be designed using the Discrete Fourier Transform (DFT) such as a Fast Fourier Transform (FFT) chip or a Recursive DFT using Residue Number Arithmetic

[9]. In the instance of section 1.3 an N-th order m-notch adaptive in-line notch filter would be designed using one of the many FIR design techniques. This could simply be an adaptive linear combiner (ie: tapped-delay-line) type FIR filter or some more sophisticated adaptive design. In the case of section 1.4 we would use the DFT implemented with an FFT chip or a recursive DFT to produce a series of band-pass filters and the adaptive algorithm would simply adapt the weights of the output subtracter that would cancel the various sine waves.

Figure 1-2 shows an implementation of the higher-order FIR filter of section 1.4 using the recursive DFT. The order of the FIR filter is set by the initial delay z^{-N} which produces N equally spaced zeros around the unit circle. The m recursive pole producing sections at the output of the delay z^{-N} are used to cancel m of these zeros resulting in m band-pass filters. These band-pass filters in turn cancel the narrowband interference from the input signal. The adaptive algorithm uses a gradient search to find the correct zeros to cancel in order to minimize the output power thus cancelling all of the narrow-band interference.

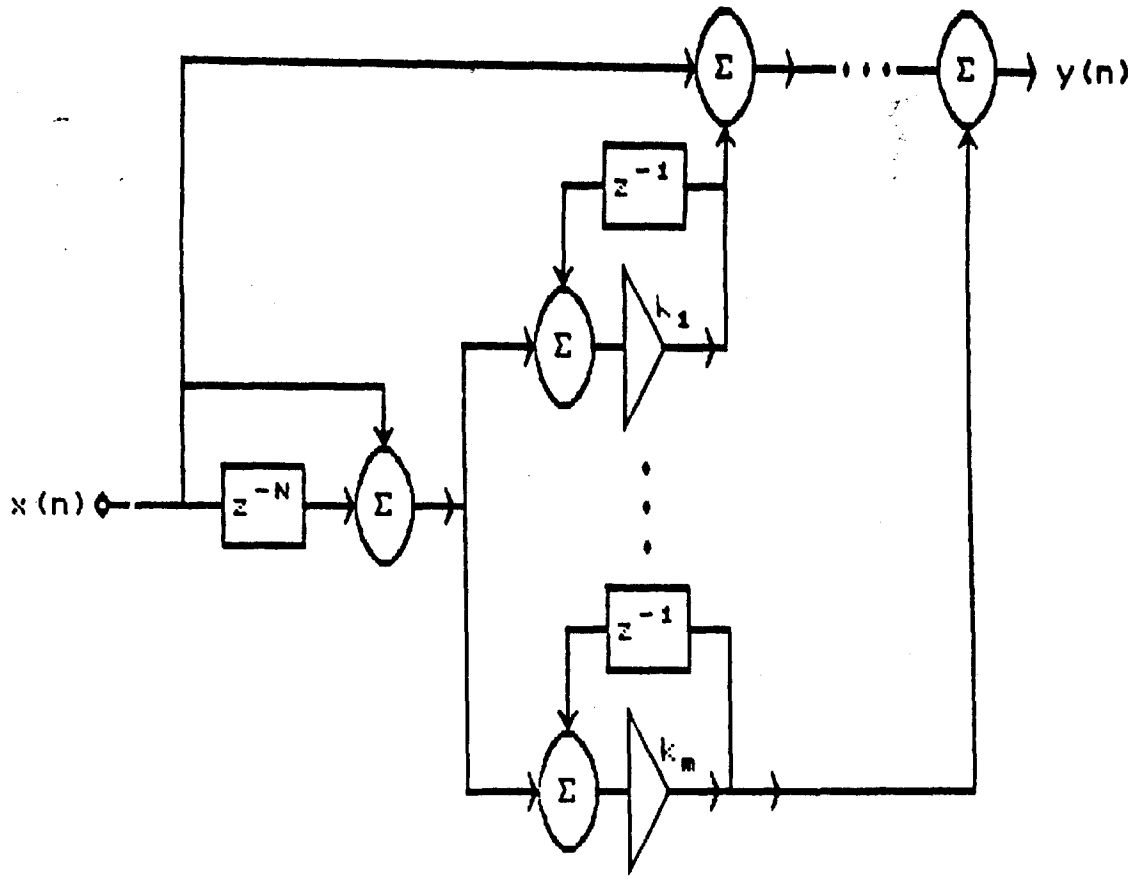


Figure 1-2. Adaptive Recursive DFT FIR Notch Filter

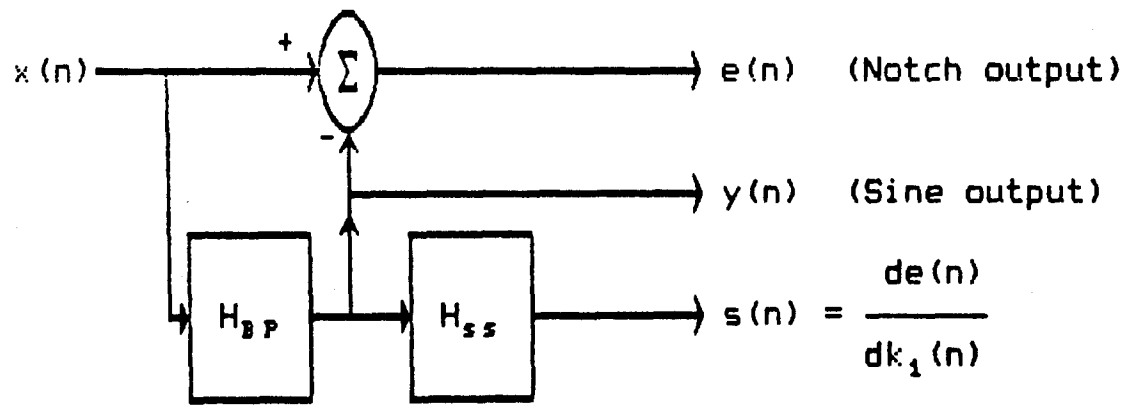


Figure 1-3. Adaptive IIR Signal Canceler

2.0 Superiority of the Second-Order Bandpass Canceler

The second-order implementations of section 1.1 and 1.2 offer considerable advantage both in hardware complexity and in adaptive performance when compared to the higher-order realizations of sections 1.3 and 1.4. A tapped-delay-line type in-line notch filter such as that of section 1.3 would require over 100 adaptive weights to perform as well as a 10-weight second-order filter. Similarly, a higher-order band-pass canceler such as that of section 1.4 would require over 100 weights to accomplish comparable performance.

The only higher-order section that appears to be competitive with a second-order system is that of Figure 1-2. If this DFT approach is used, a delay of N>100 would be required in Figure 1-2, but the adaptive part of the filter can be made comparable to that of a second-order system. In terms of performance, the DFT approach of Figure 1-2 will create a small amount of passband ripple in the resulting output signal y, but as long as N is kept large, this ripple should not be a problem. Although we believe that this approach deserves further attention, our experiments indicate that the second-order band-pass canceler approach is the best choice for our present applications. However, future developments in real-time DFT hardware or developments in such areas as multi-level Residue Number Arithmetic hardware for the circuit of Figure 1-2 might make this approach attractive.

2.1 Cascaded Second-Order Adaptive Notch Filters

The filter of section 1.1 was implemented and tested to see how it performed in the elimination of sine waves in the presence of Gaussian noise. Initial tests indicated that the cascade approach performed very well. However, when two sine waves are close together, a problem arises. Figure 2-1a shows the input used. This input consists of 3 sine waves at $0.1f_s$, $0.125f_s$, and $0.375f_s$ (36°, 45°, and 135°) in Gaussian noise. Each sine wave and the noise have the same energy in the signal (sine waves are amplitude 1.414 and the noise is Gaussian with variance 1.0). Figure 2-1b shows the output of the first stage of the notch filter, Figure 2-1c shows the output of the second stage, and Figure 2-1d shows the output of the third stage of the filter. Instead of eliminating completely one sine wave, the first

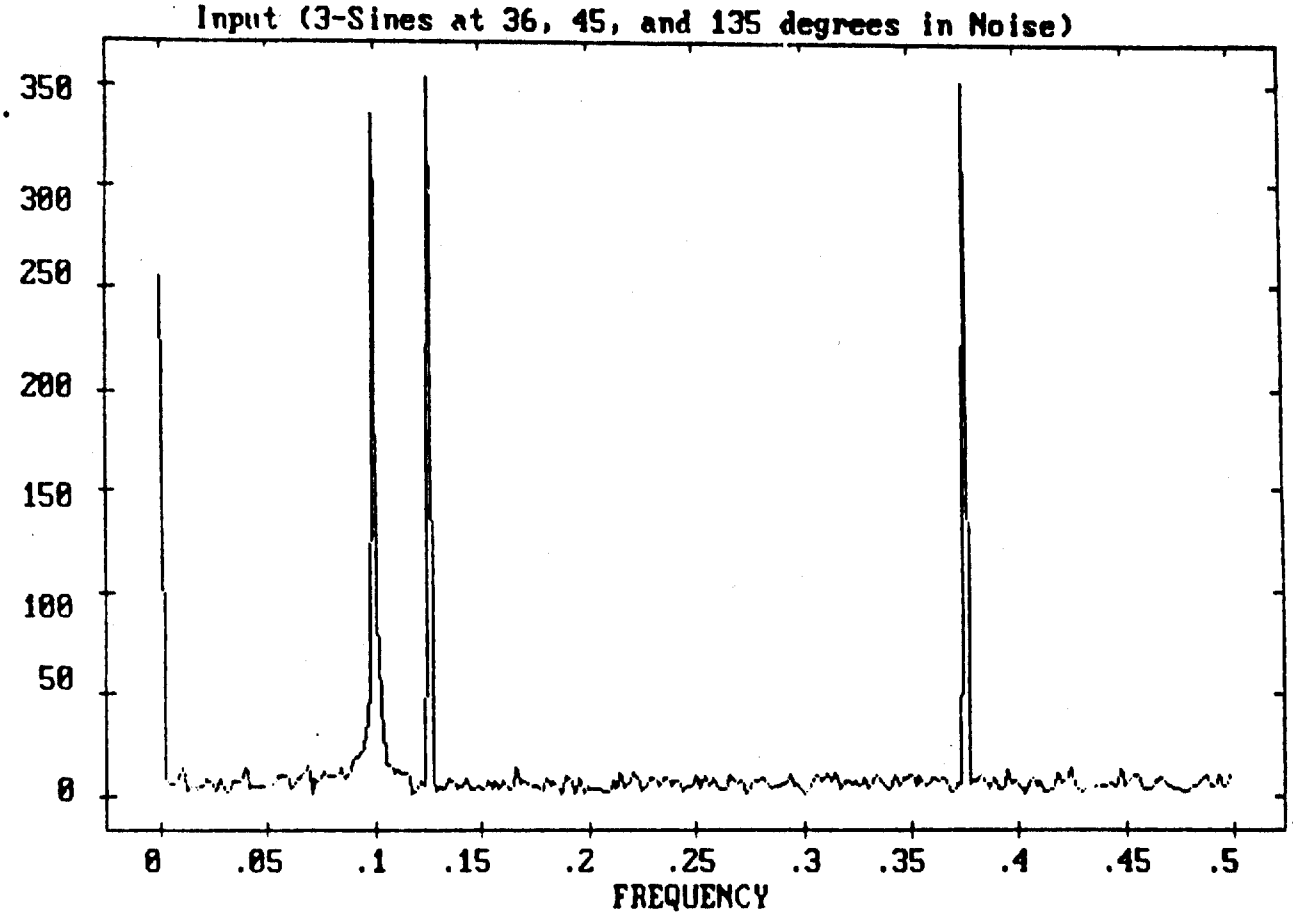


Figure 2-1a. Input to 2nd-Order Cascaded Adaptive Notch

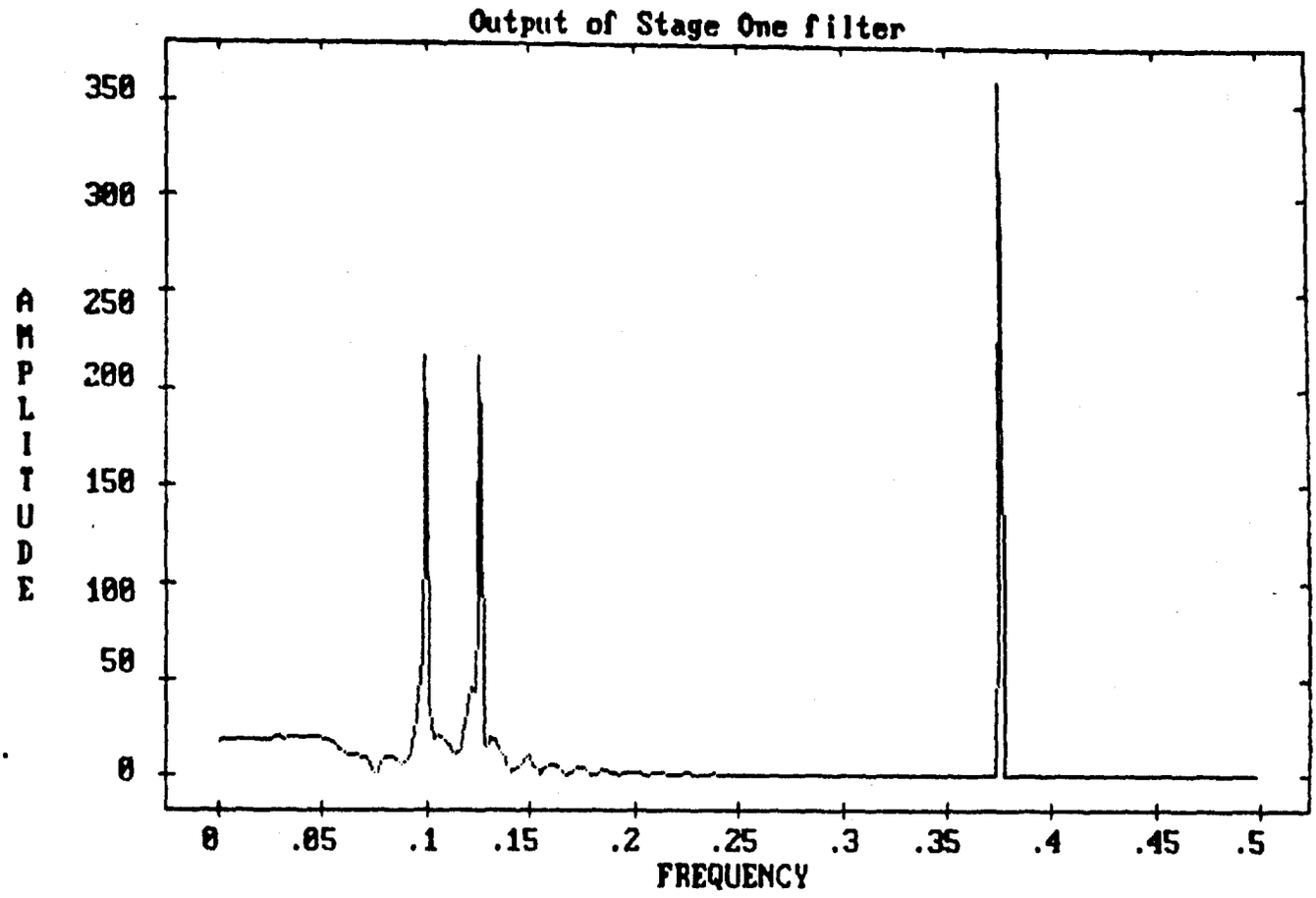


Figure 2-1b. First Stage of 2nd-Order Cascaded Adaptive Notch

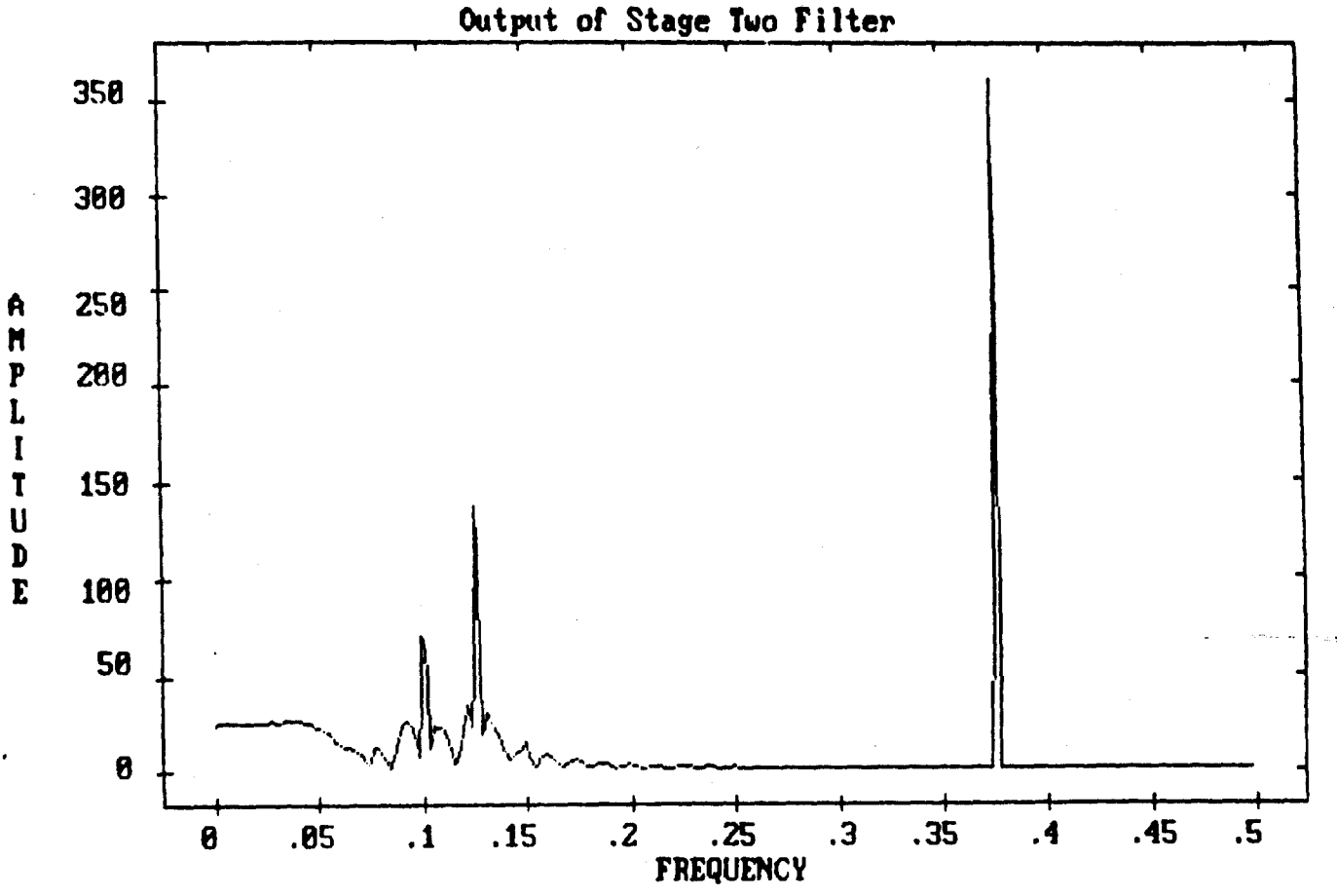


Figure 2-1c. Second Stage of 2nd-Order Cascaded Adaptive Notch

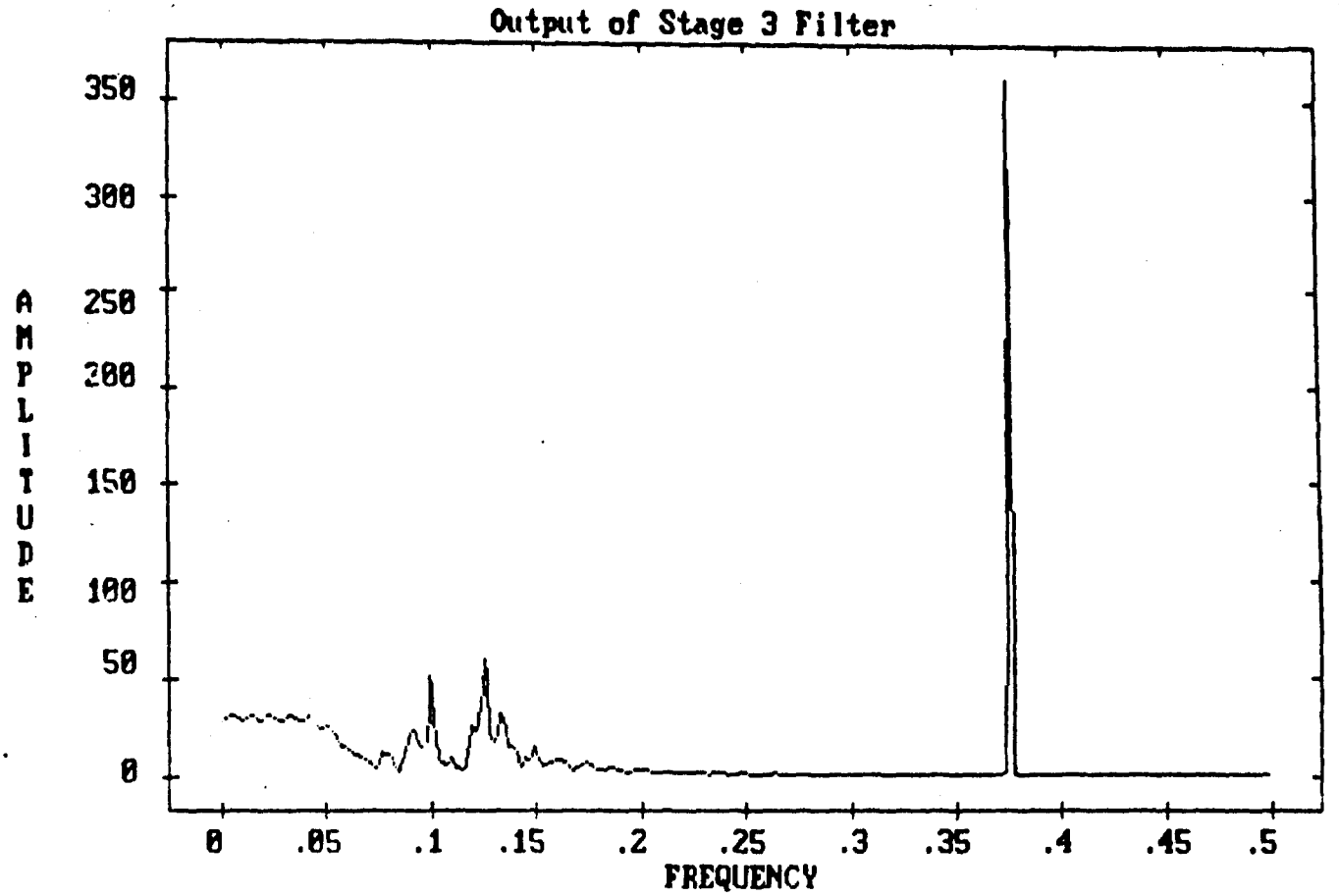


Figure 2-1d. Third Stage of 2nd-Order Cascaded Adaptive Notch

section of the notch filter (shown in Figure 2-1b) centers itself half way between the two sine waves rather than eliminating one of the sines. Then the second and third section each remove one of the sines leaving the third sine wave untouched.

2.2 The Kwan and Martin Filter

In a recent paper by Kwan and Martin [5], the problem of detecting and enhancing sinusoidal signals in the presence of noise is addressed with a cascade of IIR adaptive notch filters which are used to eliminate the sinusoids. Each of the sinusoids is eliminated by a bandpass filter whose output is an enhanced version of one of the sinusoids. Hence this remarkable structure can perform both tasks with a single adaptive filter configuration which is shown to be highly robust and performs extremely well.

The major disadvantage of the Kwan and Martin structure is that the number of biquad sections needed in the adaptive filter configuration is given by $N(N+3)/2$, where N is the number of sinusoids to be detected and removed. This becomes impractical in real-time situations with more than 4 sinusoids due to the geometric increase in the required hardware. In this section, we propose a modification to the Kwan and Martin structure that reduces the required hardware to $3N-1$ biquads with no effect on the performance of the system.

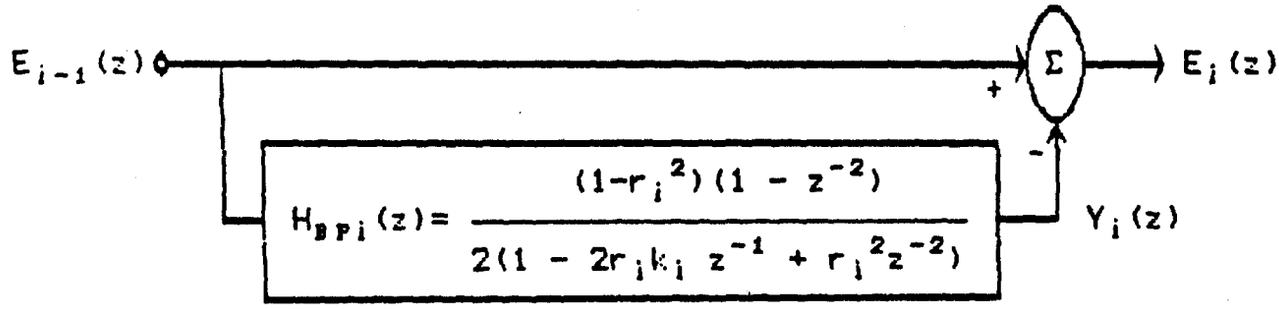


Figure 2-2. One Section of Kwan and Martin Notch Filter.

2.3 Kwan and Martin Structure

The Kwan and Martin structure consists of a cascade of IIR notch filters one stage of which is shown in Figure 2-2. Each stage consists of a bandpass filter with zeros at DC and the Nyquist frequency and unity gain at its peak frequency ω_i. Such a filter would have the following z-domain transfer function:

$$H_{BPi}(z) = \frac{1-r_i^2}{2} \frac{1 - z^{-2}}{1 - 2r_i\cos\theta_i\, z^{-1} + r_i^2 z^{-2}} \qquad (2\text{-}1)$$

where:

r_i = pole radius of the i-th section
$\theta_i = 2\pi\omega_i/\omega_s$
ω_i = peak frequency of the i-th section
ω_s = sampling frequency

Kwan and Martin identify two different methods for adapting the filter. Most of their derivation is based on what they call the "constant bandwidth" filter in which the pole radius r_i is a constant and only the frequency ω_i is adapted. An alternative approach which keeps a constant Q is also discussed in Kwan and Martin. Either of these approaches may be used with the structure we will proposing, although we too have chosen to concentrate on the constant bandwidth case in our discussion. In addition, Kwan and Martin select the adaptive quantity in such a way that it is fairly easy to determine the notch frequency from the adaptive parameter. For simplicity, we have not followed this approach, but have chosen the adaptive parameter $k_i=\cos\theta_i$. Substituting this into equation (2-1) yields the following:

$$H_{BPi}(z) = \frac{1-r_i^2}{2} \frac{1 - z^{-2}}{1 - 2r_ik_i\, z^{-1} + r_i^2 z^{-2}} \qquad (2\text{-}2)$$

From Figure 2-2, we see that the notch filter for each section is the difference between 1 and the bandpass filter, hence:

$$H_{Ni}(z) = 1 - H_{BPi}(z) \qquad (2\text{-}3a)$$

$$H_{Ni}(z) = \frac{1+r_i^2}{2} \frac{1 - 4k_i r_i z^{-1}/(1+r_i^2) + z^{-2}}{1 - 2r_i k_i z^{-1} + r_i z^{-2}} \qquad (2\text{-}3b)$$

CALCULATION OF THE GRADIENT

The basic structure of the Kwan and Martin adaptive filter shown in Figure 2-3 is a cascade of N sections of the form of Figure 2-2.

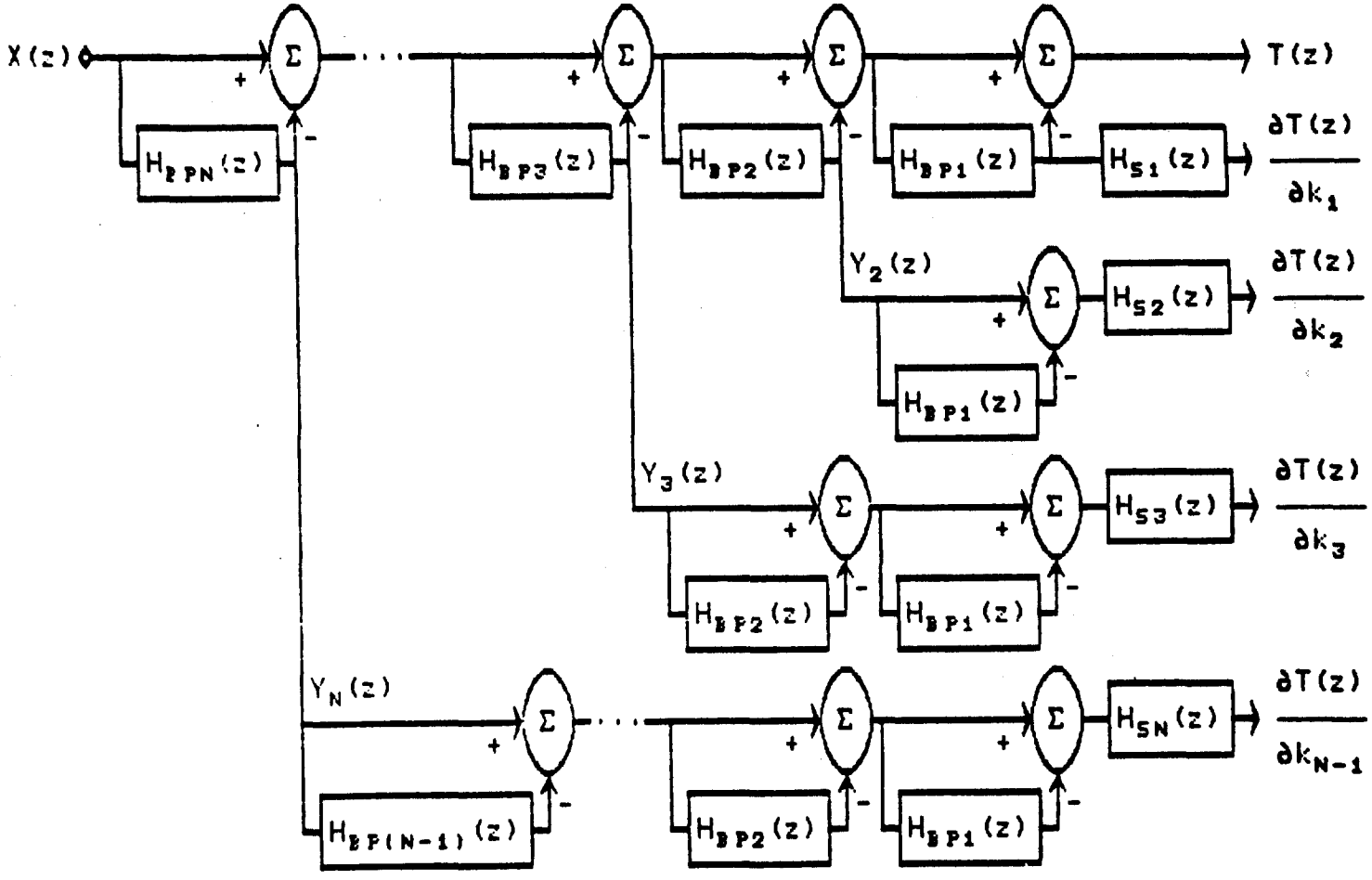


Figure 2-3. Kwan and Martin Notch Filter for Tracking N sinusoids

The overall transfer function is given by:

$$T(z) = \prod_{i=1}^{N} H_{Ni}(z) \qquad (2\text{-}4a)$$

$$= \prod_{i=1}^{N} (1 - H_{BPi}) \qquad (2\text{-}4b)$$

Kwan and Martin choose as their objective function J(z) the square of the output of the final stage of the cascade:

$$J(z) = [E_1(z)]^2 = T(z)^2 X(z)^2 \qquad (2\text{-}5)$$

Hence, the gradient of the objective function J(z) is given by:

$$\frac{\partial J(z)}{\partial k_j} = 2E_1(z)X(z)\frac{\partial T(z)}{\partial k_j} \qquad (2\text{-}6)$$

Thus in order to find the gradient with respect to the adaptive parameters k_j, we must take the partial derivative of T(z) with respect to each k_j:

$$\frac{\partial T(z)}{\partial k_j} = \prod_{\substack{i=1 \\ i \neq j}}^{N} H_{Ni}(z) \frac{\partial H_{Nj}(z)}{\partial k_j} \qquad (2\text{-}7)$$

From equation (2-3a) we have:

$$\frac{\partial H_{Nj}(z)}{\partial k_j} = \frac{\partial [1 - H_{BPj}(z)]}{\partial k_j} = \frac{-\partial H_{BPj}(z)}{\partial k_j} \qquad (2\text{-}8)$$

From equation (2-2) we have:

$$\frac{\partial H_{Nj}(z)}{\partial k_j} = -H_{BPj}(z)\,\frac{-2r_j z^{-1}}{D(z)} = H_{BPj}(z)H_{Sj}(z) \qquad (2\text{-}9)$$

where:

$$D(z) = 1 - 2r_j k_j z^{-1} + r_j^{-2} \quad \text{[denominator of } H_{BPj}(z)\text{]}$$

$$H_{Sj}(z) = \frac{2r_j z^{-1}}{1 - 2r_j k_j z^{-1} + r_j^{-2}} = \frac{2r_j z^{-1}}{D(z)}$$

Substituting equation (2-9) into equation (2-7) we obtain:

$$\frac{\partial T(z)}{\partial k_j} = \prod_{\substack{i=1 \\ i \neq j}}^{N} H_{Ni}(z)\; H_{BPj}(z)H_{Sj}(z) \qquad (2\text{-}10)$$

Figure 2-3 shows the Kwan and Martin realization of the complete adaptive system. The difficulty is in generating the product of notch filters without the notch filter "j", which is required in equation (2-10). To generate this product for each section, would require N-1 biquads per section resulting in a total of N(N-1) biquads just to generate the product. Kwan and Martin are able to reduce this by using the output of the bandpass filter as the input to their cascade. Since this output already has (j-1) of the required $H_{Ni}(z)$ factors in it, only (N-j) additional biquads are needed for a total of:

$$\sum_{j=1}^{N} (N - j) = \sum_{j=1}^{N} N - \sum_{j=1}^{N} j = N^2 - \frac{N^2+N}{2} = \frac{N^2-N}{2} \qquad (2\text{-}11)$$

Adding this to the N biquads required to realize the cascade of notch filters and the N biquads required to realize the $H_{sj}(z)$ factors yields a total number of biquads given by:

$$\#\text{ of biquads (Kwan/Martin)} = \frac{N^2-N}{2} + 2N = \frac{N(N+3)}{2} \qquad (2\text{-}12)$$

2.4 Improved Structure

Figure 2-4 shows the improved adaptive notch filter structure proposed in this paper. The key to the improvement is the recognition that the output $E_1(z)=T(z)X(z)$ for the cascade of the notch filters can be written both as a product of the individual notch filter section transfer functions $H_{Ni}(z)$ time the input and in terms of the input $X(z)$ minus the outputs $Y_i(z)$ of the bandpass filters:

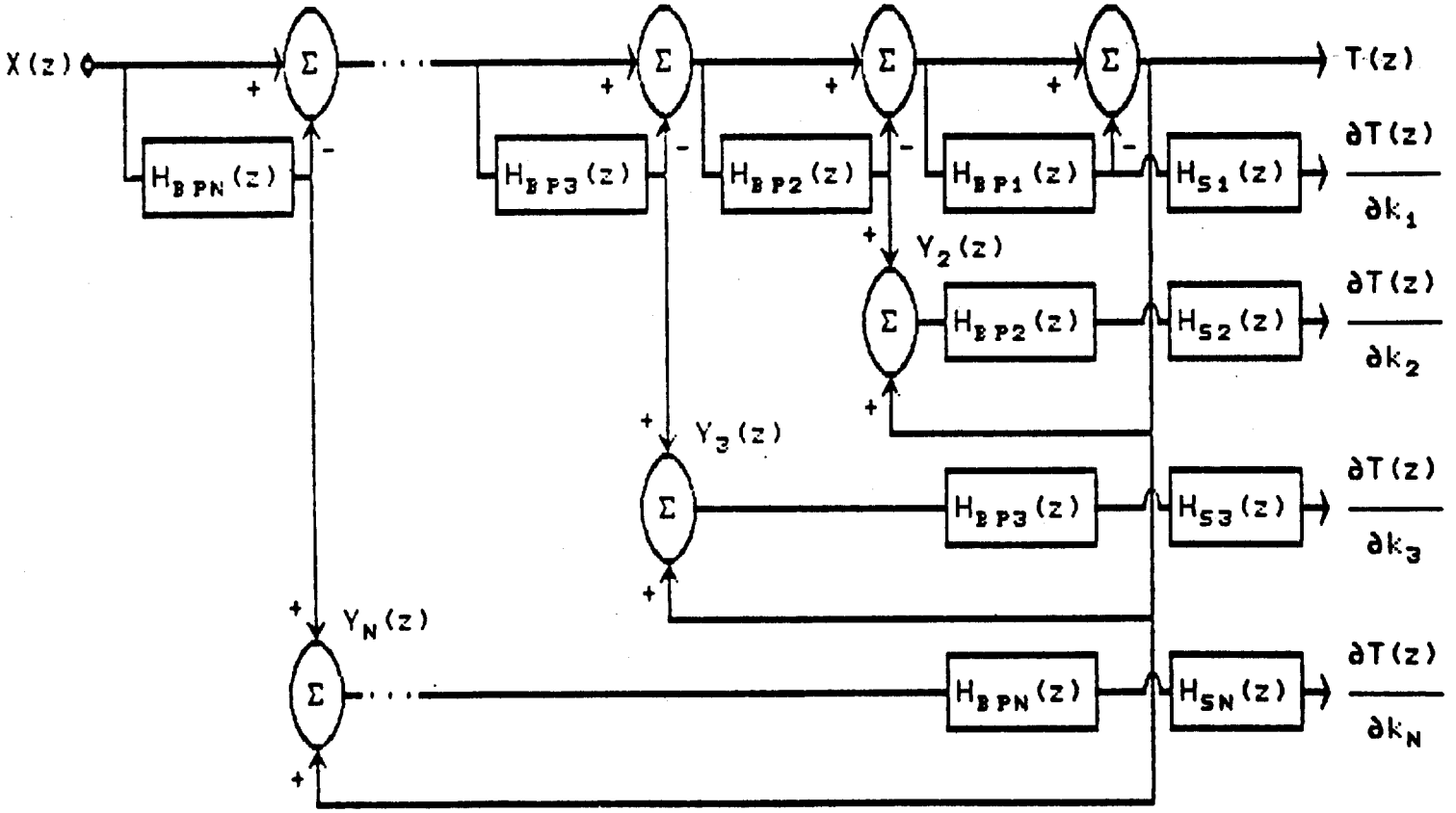


Figure 2-4. New Notch Filter for Tracking N sinusoids

$$E_1(z) = T(z)X(z) = \prod_{i=1}^{N} H_{Ni}(z)\ X(z) \qquad (2\text{-}13a)$$

$$= X(z) - \sum_{i=1}^{N} Y_i(z) \qquad (2\text{-}13b)$$

To get the product of $H_{Ni}(z)$ without the term i=j, we may use equation (2-13b) to simply add back the term $Y_j(z)$:

$$\prod_{\substack{i=1 \\ i \neq j}}^{N} H_{Ni}(z)\ X(z) = T(z)X(z) - Y_j(z) \qquad (2\text{-}14)$$

Figure 2-4 makes use of this fact to generate the gradient needed for the adaptive process. From the Figure, we can see that the total number of biquads required is N for the cascade of notch filters plus 2N for the $H_{BPi}(z)H_{Si}(z)$ required for adaptation, minus 1 at the last stage, since the last stage does not need the extra $H_{BPi}(z)$. Thus we have:

$$\#\text{ of biquads (new structure)} = 3N-1 \qquad (2\text{-}15)$$

Table 2-I shows a comparison of the number of biquads required for the Kwan and Martin structure vs the number required for the new structure for various values of N. As can be seen from the table, the two structures are very competitive for N up to 4, but beyond N=4 the new structure offers a considerable hardware savings.

Table 2-I

N	Kwan and Martin Filter N(N+3)/2	Proposed Filter 3N-1
1	2	2
2	5	5
3	9	8
4	14	11
5	20	14
6	27	17
7	35	20
8	44	23
9	54	26
10	65	29

2.5 Experimental Results Using the New Structure

The algorithm was compared to the Kwan and Martin algorithm and found to perform as well or better than Kwan and Martin in all of the experiments. Figure 2-5 shows an example of the new algorithm with 5 notches applied to a signal with 5 sine waves imbedded in noise. Figure 2-5a shows the input signal frequency response, Figure 2-5b shows the output after filtering and Figure 2-5c shows the frequency response of the output after filtering. Figure 2-5d shows the adaptation of the five parameters. Due to computational complexity, we were unable to do 5 notches using the Kwan and Martin algorithm, thus just being able to implement the new algorithm represents an improvement over Kwan and Martin.

Figure 2-6 shows what happens when the 5-notch adaptive filter is used when there are only 3 sines imbedded in noise. Figure 2-6a shows the input spectrum, Figure 2-6b shows the output, Figure 2-6c shows the output spectrum, and Figure 2-6d shows the adaptation of the 5 notches. Three of the notches cancel the three sines, while the remaining notches wander around.

Figure 2-7 shows what happens when the 5-notch adaptive filter is used when there are 7 sines imbedded in noise. Figure 2-7a shows the input spectrum, Figure 2-7b shows the output, Figure 2-7c shows the output spectrum, and Figure 2-7d shows the adaptation of the 5 notches. What happens in Figure 2-7d is very interesting and shows just how robust the algorithm is. The first 4 notches lock solidly on the first 4 frequencies (30°, 45°, 60°, and 90°). The final notch is left jumping between the remaining three frequencies (120°, 135°, and 150°). Since it is unable to cancel all three, it spends a little bit of time on each one, thus attenuating each of them. Occasionally, it will scan across the entire range in hope of finding a better solution. But it quickly settles back to the pattern of attenuating each of the remaining sines.

The final experiment we tried with the sine waves is shown in Figure 2-8. Here we used 10 notches and 10 sines in order to demonstrate the ability of the algorithm to process 10 sine waves. Figure 2-8 shows the adaptation of the 10 parameters.

2.6 Conclusion

We have demonstrated that the new adaptive notch filter structure uses considerable less hardware than the structure proposed by Kwan and Martin, yet performs the same (or better) in typical applications. In addition to the results demonstrated here, chapters 3 and 4 of this report and an M.S. Thesis [10] provide other experiments to verify the performance of the new structure. This work has also been reported at two conferences [10,11]. In all of these cases, the new structure performs as well or better than the Kwan and Martin structure.

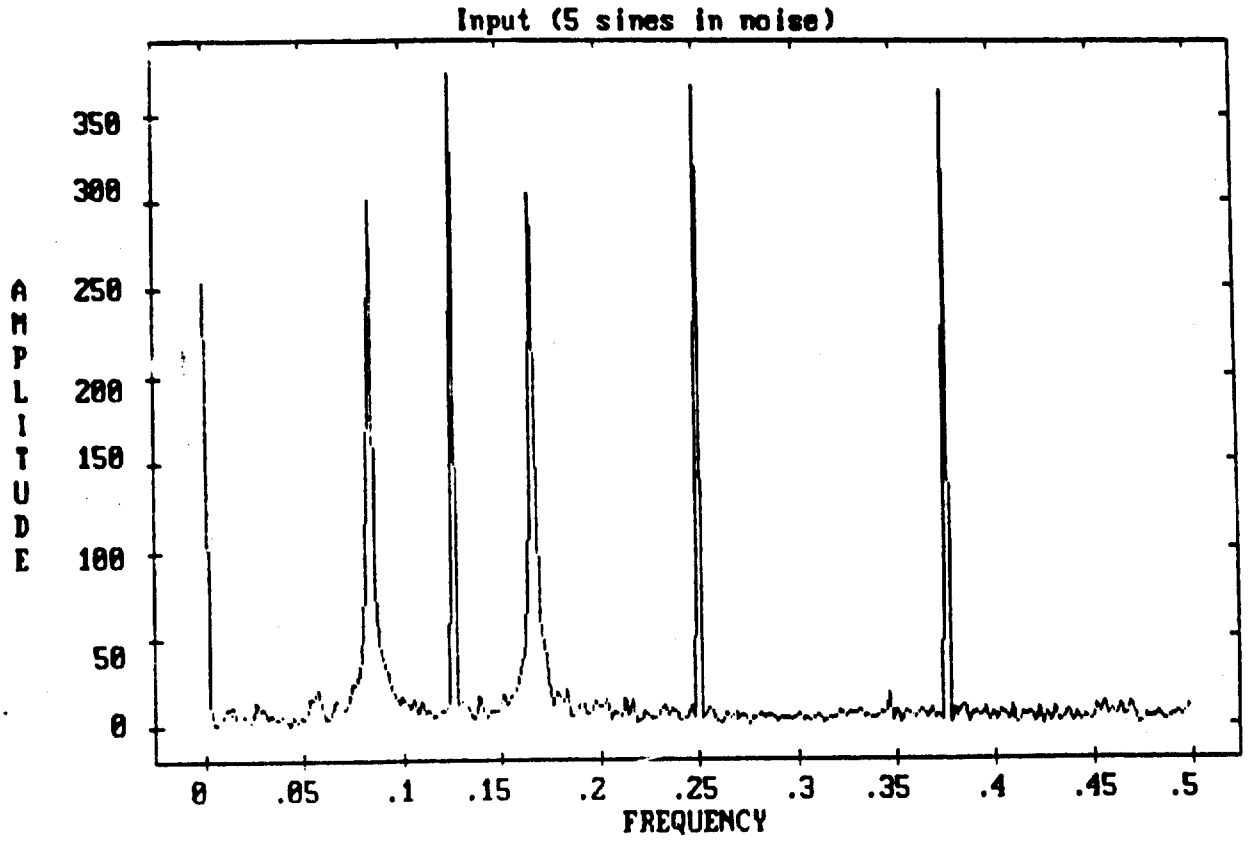


Figure 2-5a Input of 5 Sines (30°, 45°, 36°, 60°, and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 5: AMP = {1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {.083, .125, .167, .250, .375} ANG = {0.0, 0.0, 0.0, 0.0, 0.0}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

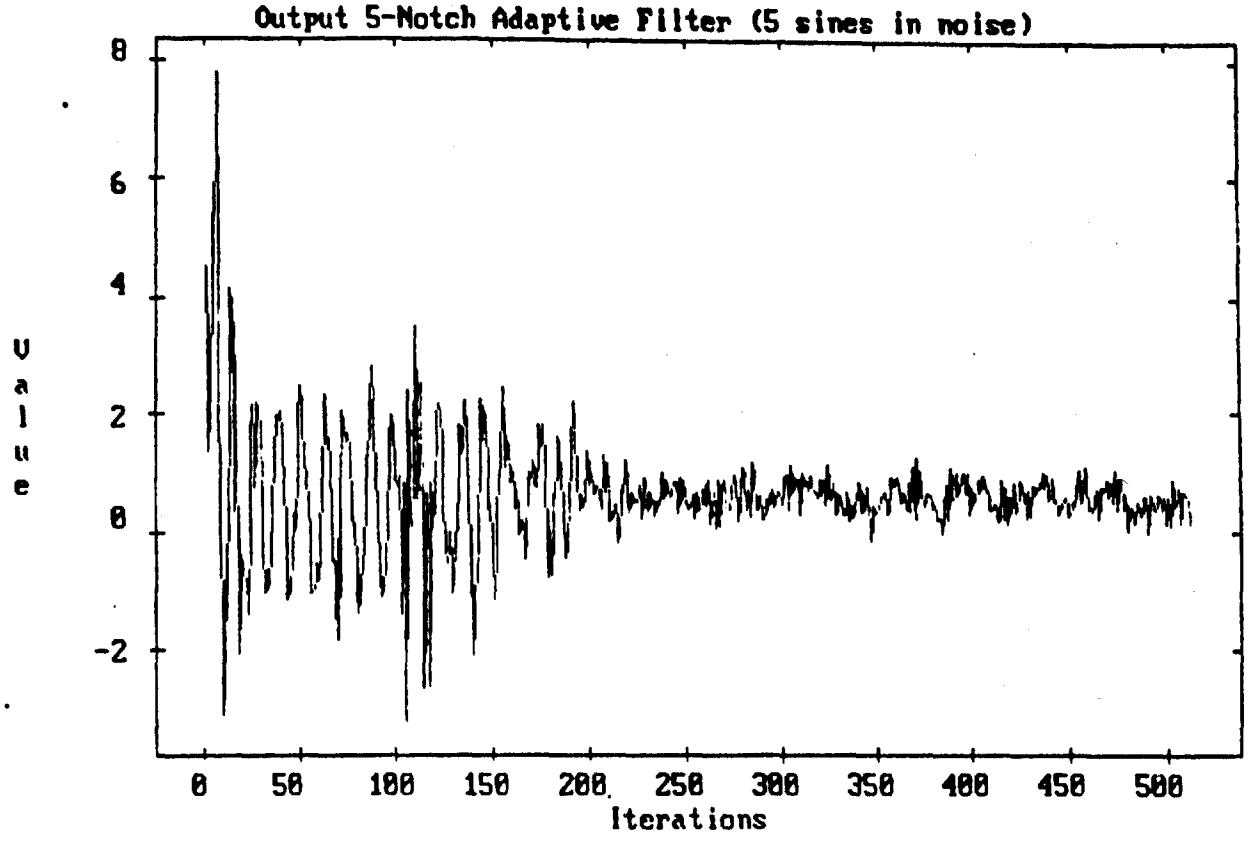


Figure 2-5b Filtered Output of 5-Notch Adaptive Filter
Input 3 Sines (30°, 45°, 36°, 60° and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 5: AMP = {1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {.083, .125, .167, .250, .375} ANG = {0.0, 0.0, 0.0, 0.0, 0.0}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

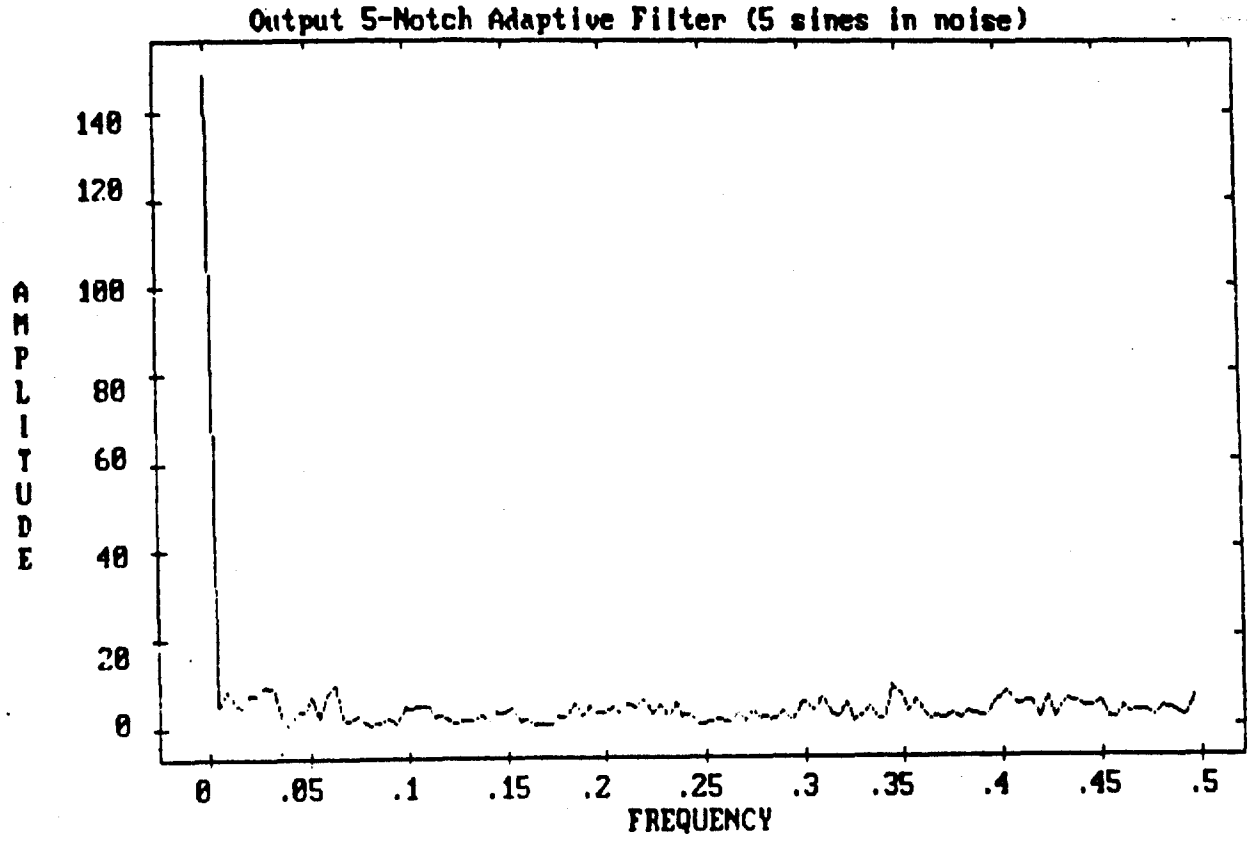


Figure 2-5c Frequency Response of Filtered Output of 5-Notch Filter Input 3 Sines (30°, 45°, 36°, 60° and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 5: AMP = {1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {.083, .125, .167, .250, .375} ANG = {0.0, 0.0, 0.0, 0.0, 0.0}
DC = 0.003, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

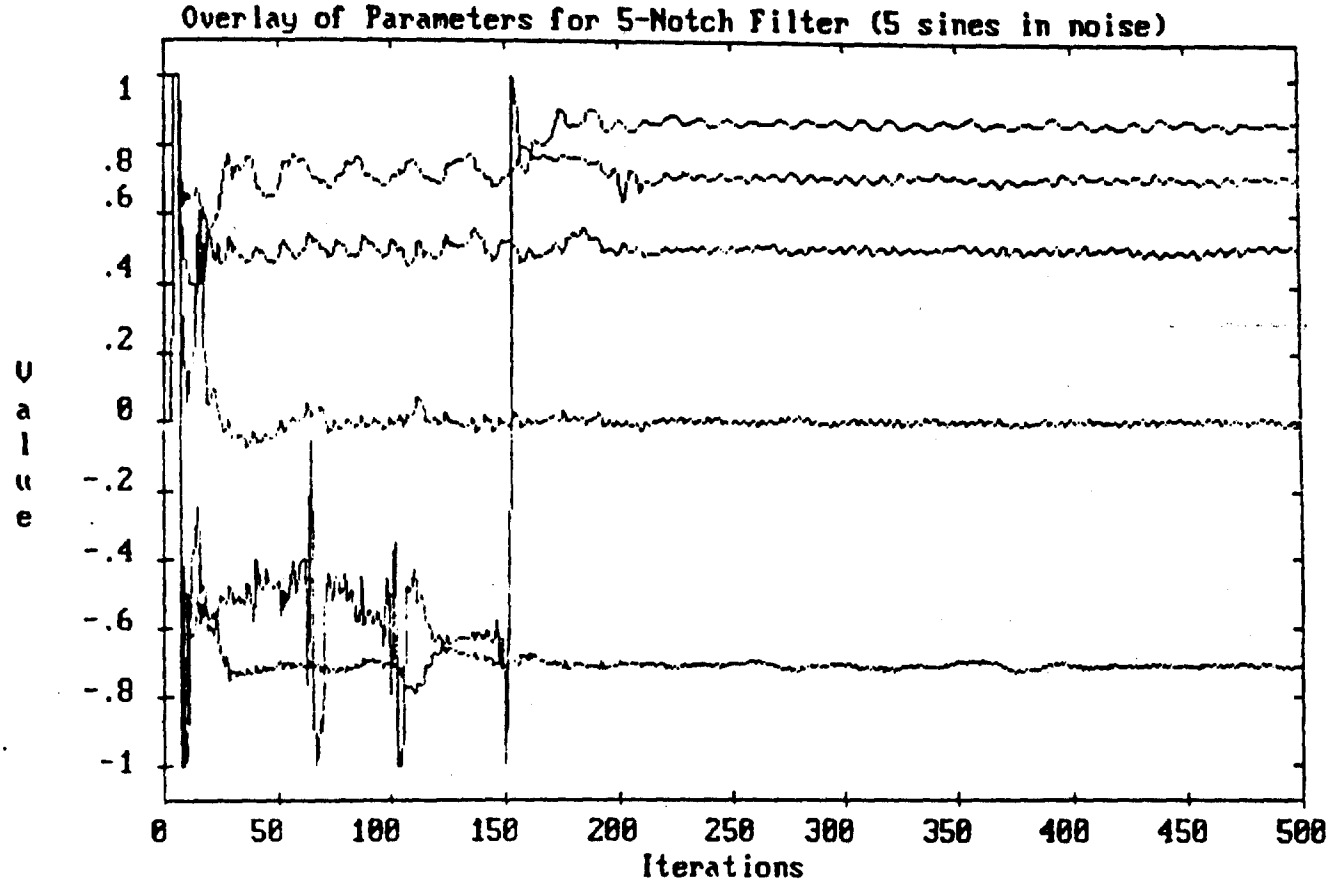


Figure 2-5d Parameter Adaptation of 5-Notch Filter
Input 3 Sines (30°, 45°, 36°, 60° and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 5: AMP = {1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {.083, .125, .167, .250, .375} ANG = {0.0, 0.0, 0.0, 0.0, 0.0}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

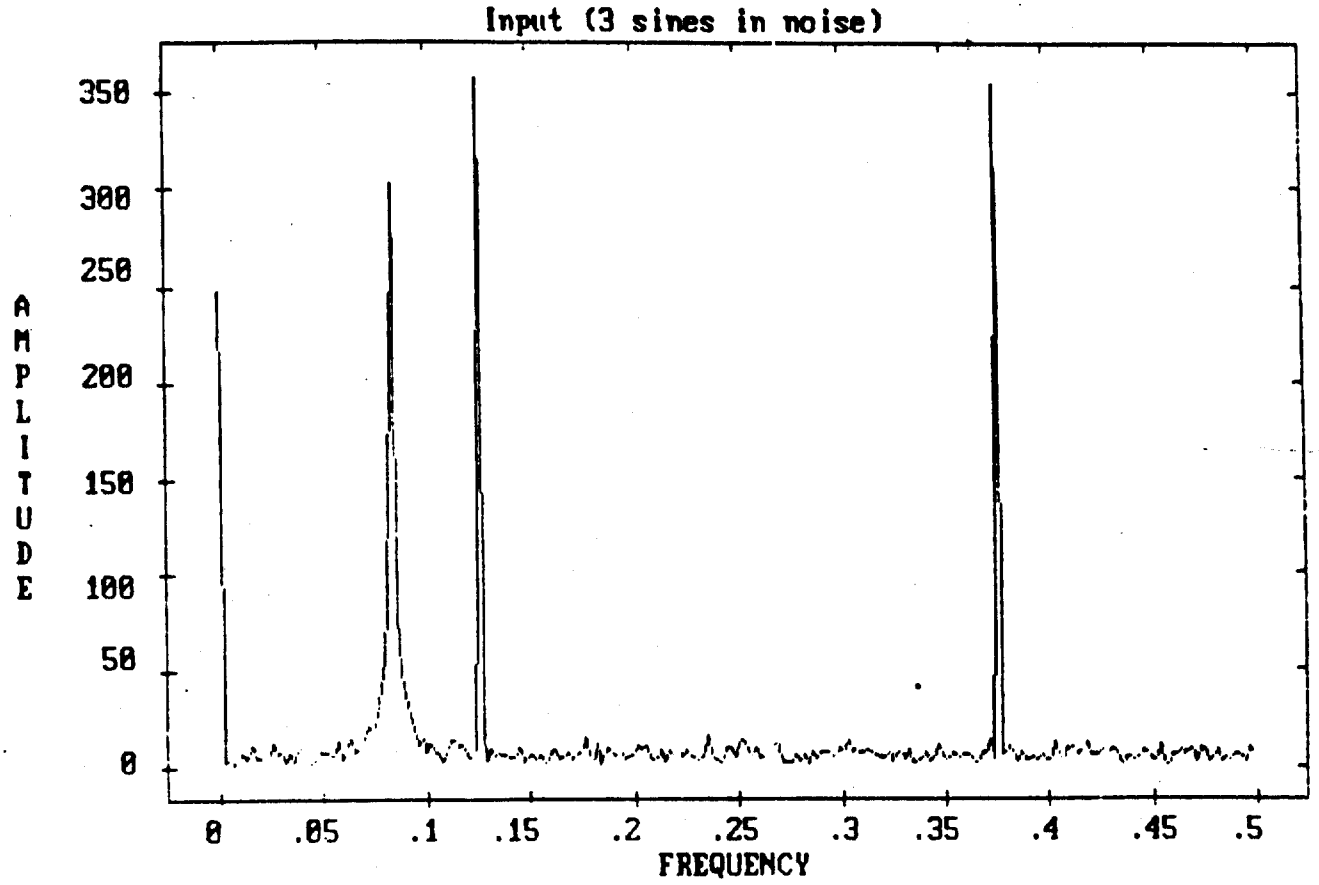


Figure 2-6a Input of 3 Sines (30°, 36°, and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 3: AMP = {1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.375} ANG = {0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

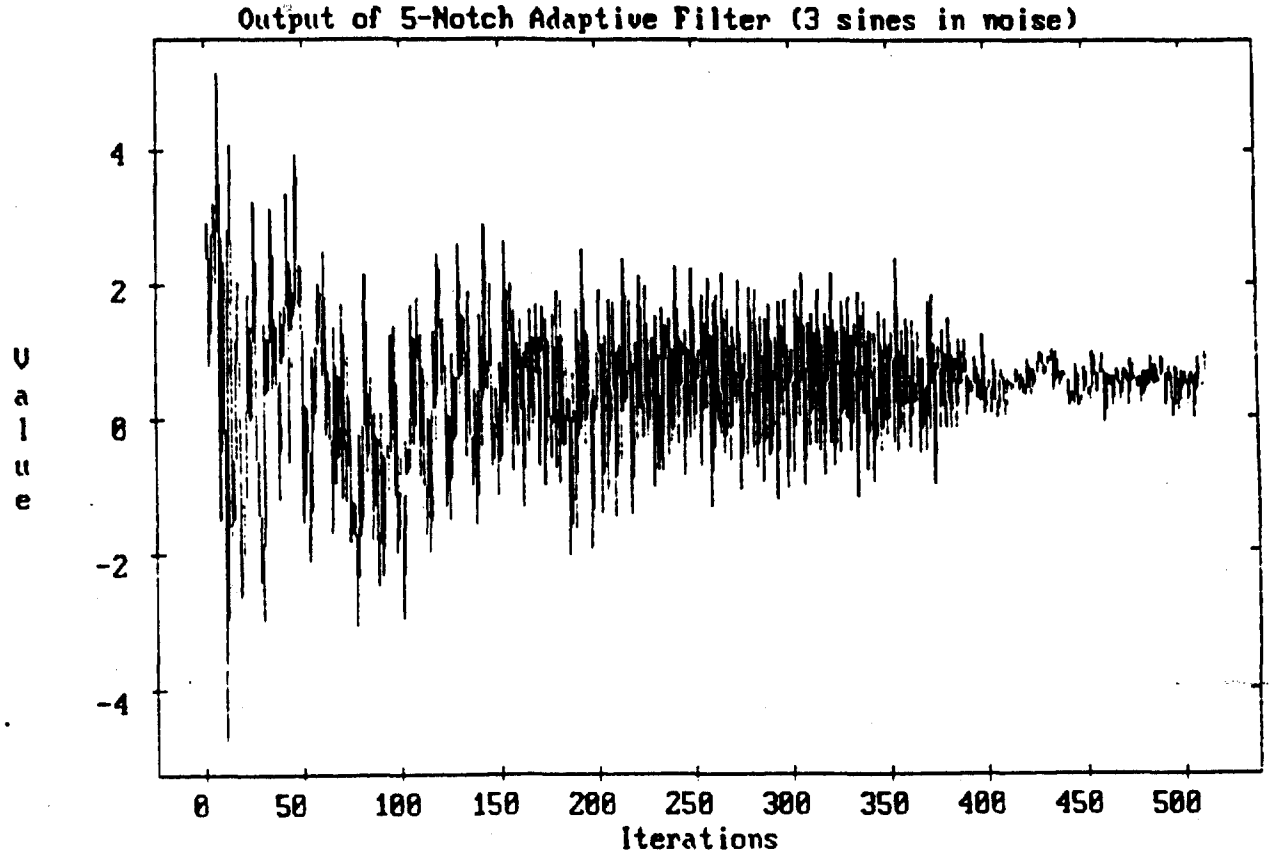


Figure 2-6b Filtered Output of 5-Notch Filter
Input 3 Sines (30°, 36°, and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 3: AMP = {1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.375} ANG = {0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

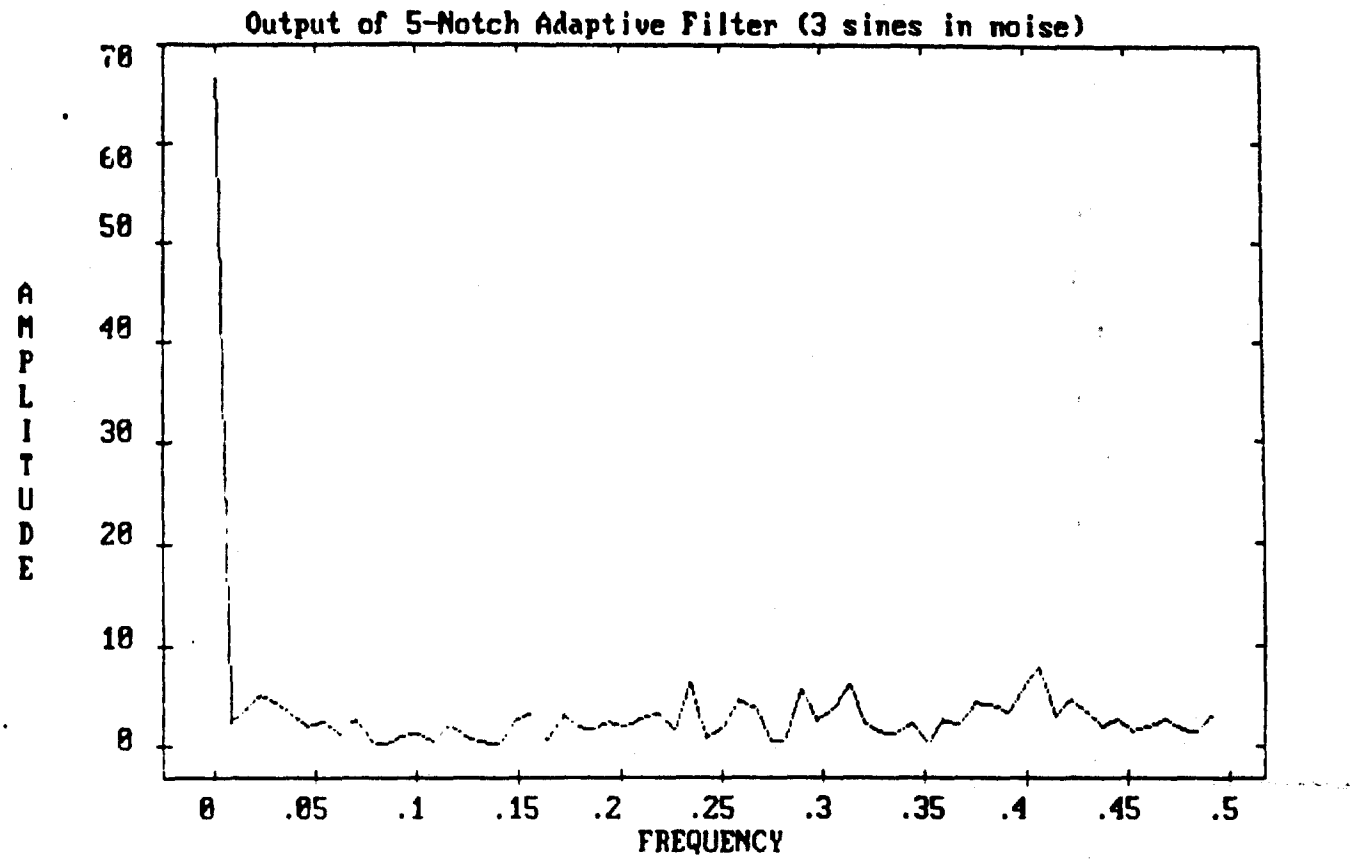


Figure 2-6c Frequency Response of Filtered Output of 5-Notch Filter Input 3 Sines (30°, 36°, and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 3: AMP = {1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.375} ANG = {0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

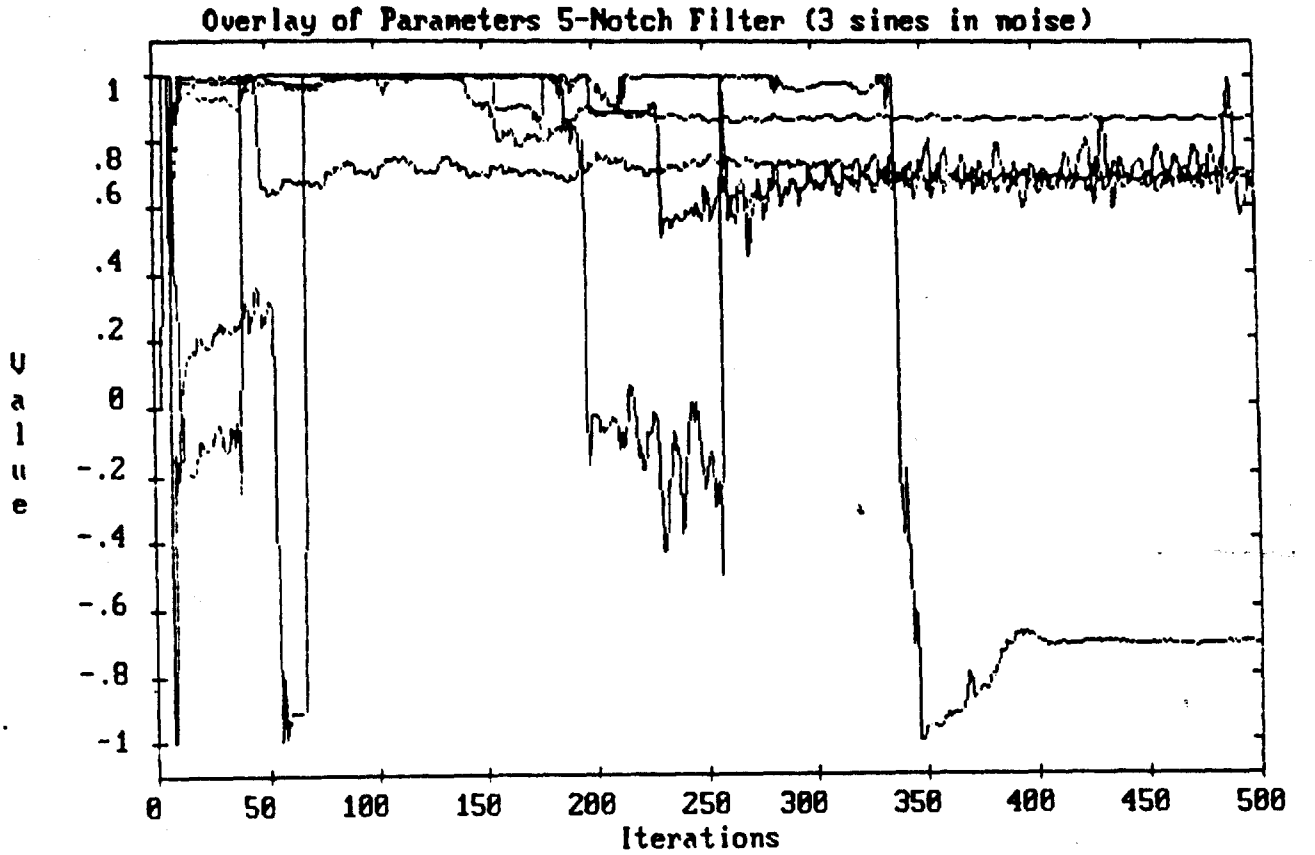


Figure 2-6d Parameter Adaptation for 5-Notch Filter
Input 3 Sines (30°, 36°, and 135°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 3: AMP = {1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.375} ANG = {0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

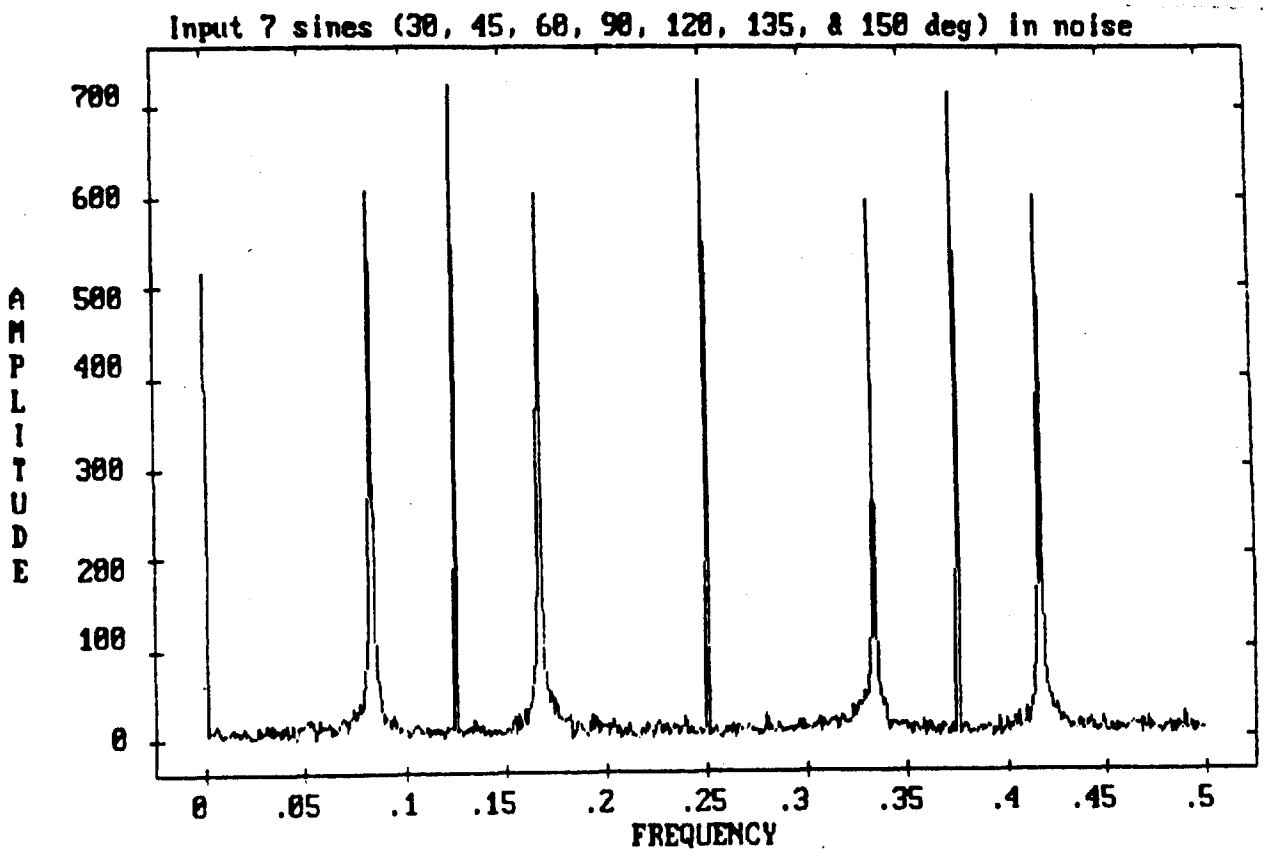


Figure 2-7a Input of 7 Sines
(30°, 45°, 60°, 90°, 120°, 135°, and 150°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 7.
AMP = {1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.167, 0.250, 0.333, 0.375, 0.417}
ANG = {0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second orcer forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

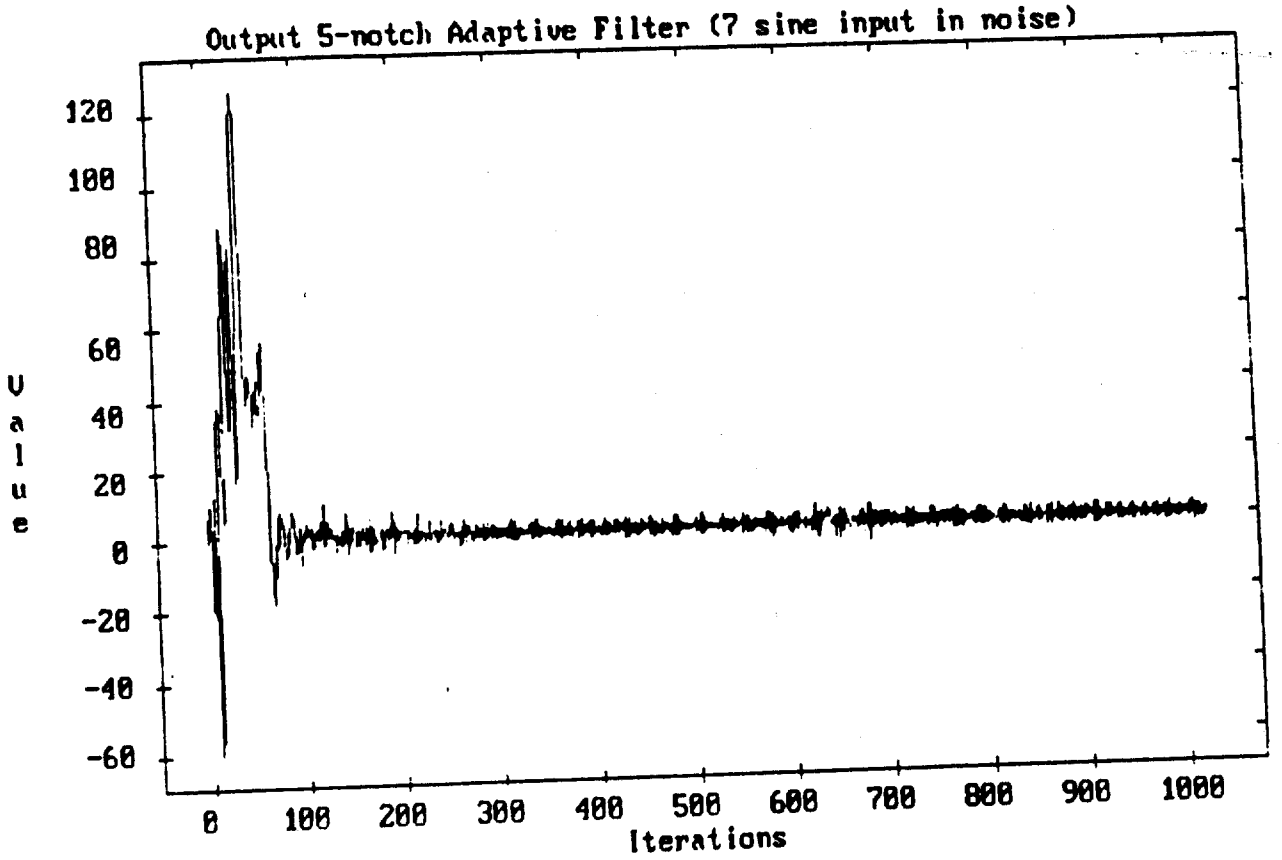


Figure 2-7b Filtered Output of 5-Notch Filter
Input 7 Sines (30°, 45°, 60°, 90°, 120°, 135°, and 150°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 7.
AMP = {1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.167, 0.250, 0.333, 0.375, 0.417}
ANG = {0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

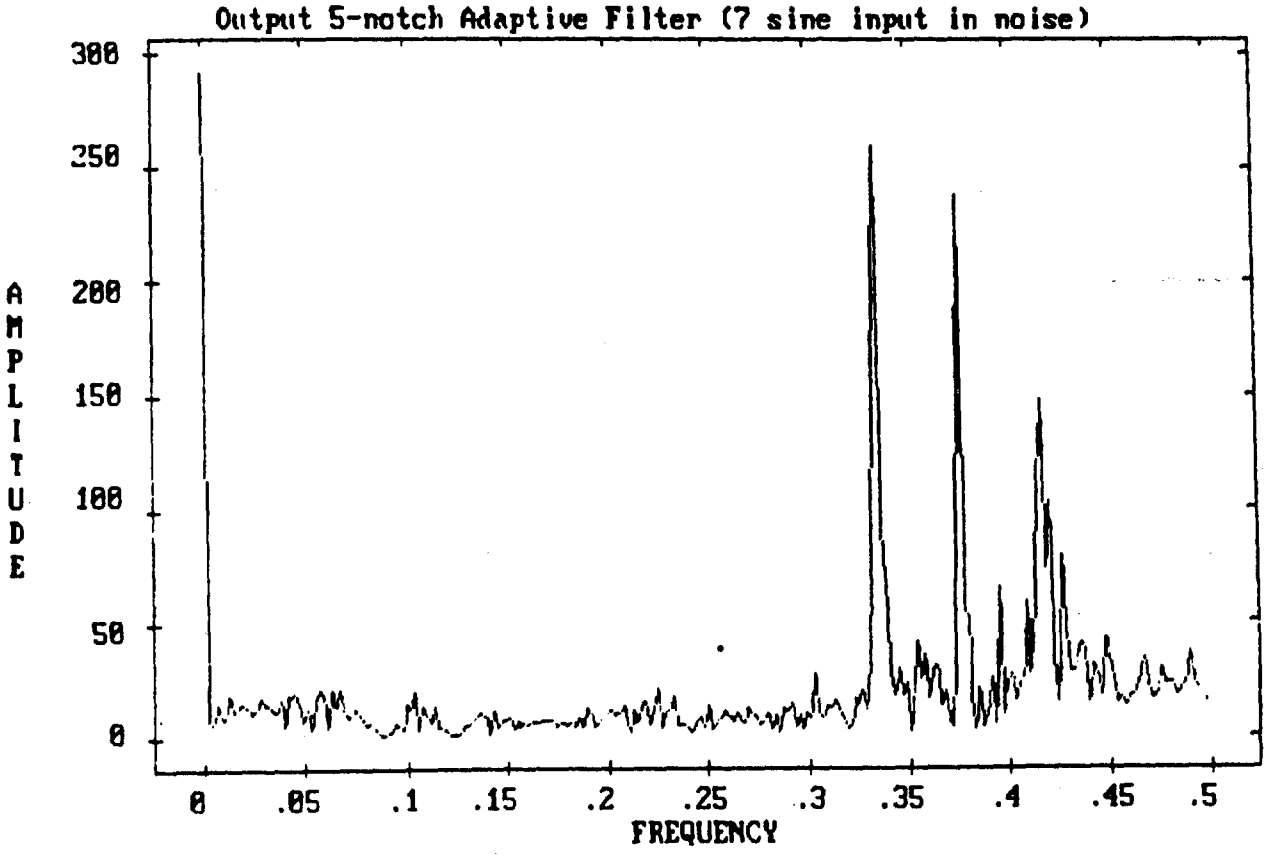


Figure 2-7c Frequency Response of Filtered Output of 5-Notch Filter Input 7 Sines (30°, 45°, 60°, 90°, 120°, 135°, and 150°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 7.
AMP = {1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.167, 0.250, 0.333, 0.375, 0.417}
ANG = {0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

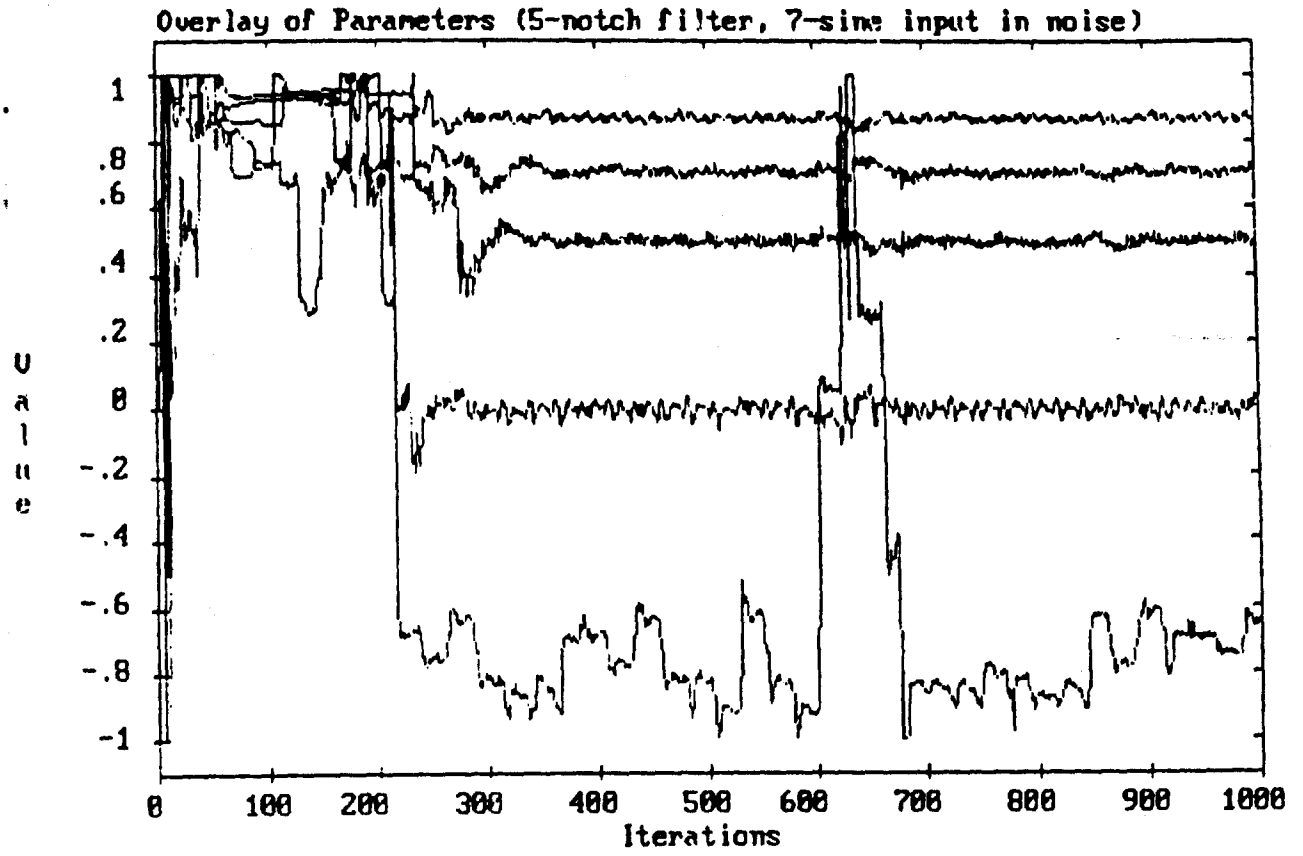


Figure 2-7d Parameter Adaptation of 5-Notch Filter
Input 7 Sines (30°, 45°, 60°, 90°, 120°, 135°, and 150°) in Noise

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 7.
AMP = {1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {0.083, 0.125, 0.167, 0.250, 0.333, 0.375, 0.417}
ANG = {0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000}
DC = 0.000, NOISE = 1.000, k_limit = 1.000, slow = 1, n_sec = 5
Initial values of k1 = {0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii = {0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor = {0.900, 0.900, 0.900, 0.900, 0.900}

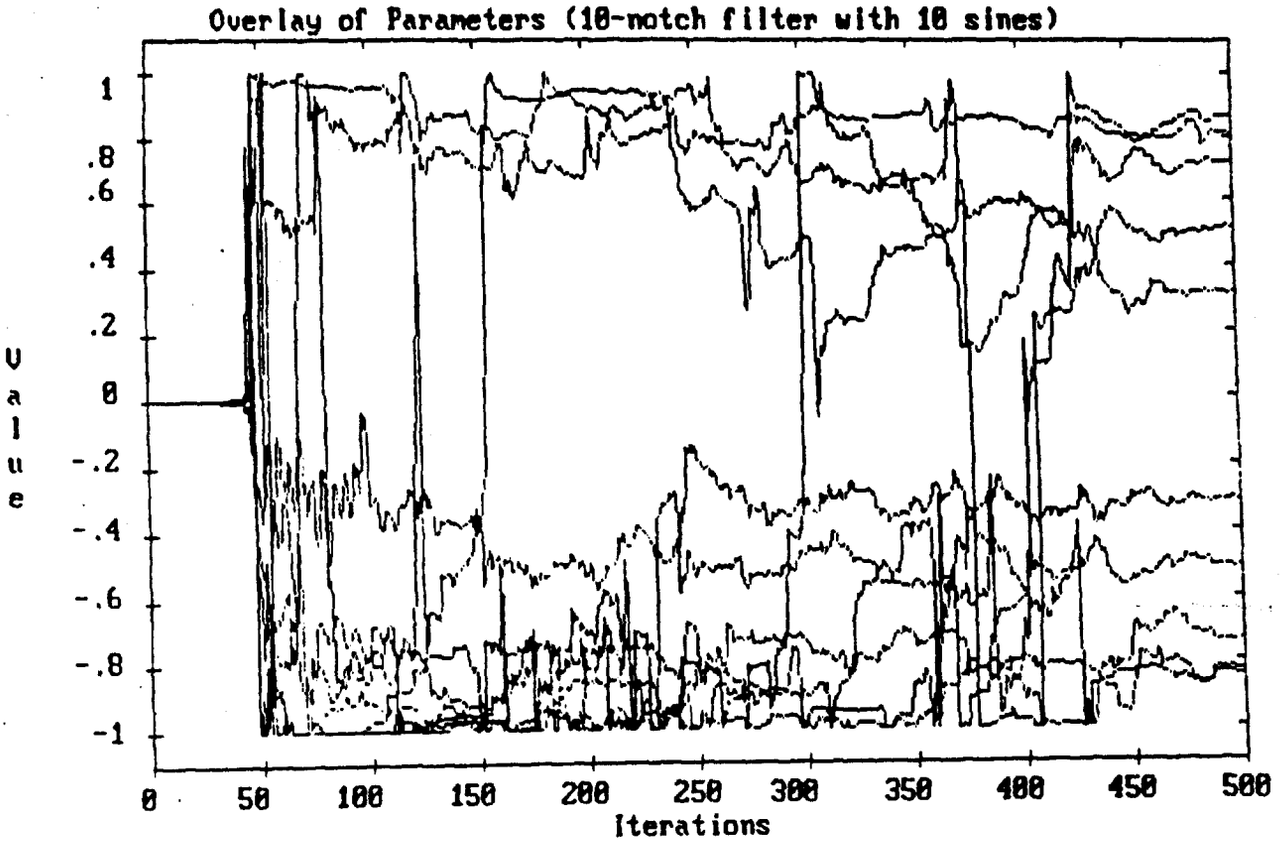


Figure 2-8 Input of 10 Sines in Noise
(30°, 36°, 45°, 60°, 72°, 108°, 120°, 135°, 144°, 150°)

step = 0.100000, s_limit = 0.000000, pmin =0.000001, pmax = 1000000.00
Number of sinusoidal inputs is 10.
AMP = {1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414, 1.414}
FREQ = {0.083, 0.100, 0.125, 0.167, 0.200, 0.300, 0.333, 0.375, 0.400, 0.417}
ANG = {0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000}
DC = 0.000, NOISE = 0.000, k_limit = 1.000, slow = 1, n_sec = 10
Initial values of k1 =
{0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000}
Notch pole radii =
{0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900}
Second order forgetting. Factor =
{0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900, 0.900}

3.0 Detailed Description of the New Algorithm

In this chapter of the report we will discuss in detail the new algorithm and demonstrate its performance on simulated data. The derivation of the structure and the gradient calculation were discussed in sections 2.3 and 2.4. Here we shall give the details for implementing the adaptation, information on how to choose the various adaptive parameters, and demonstration of the effects of different choices on the performance of the filters.

3.1 The Adaptation Algorithm

Figure 2-4 gives the structure for the new algorithm. The structure contains two types of second-order filters:

Band-pass filters (see equation 2-2):

$$H_{BPi}(z) = \frac{1-r_i^2}{2} \frac{1 - z^{-2}}{1 - 2r_i k_i z^{-1} + r_i^2 z^{-2}} \qquad (3\text{-}1)$$

and Sensitivity filters (see equation 2-9):

$$H_{Sj}(z) = \frac{2r_j z^{-1}}{1 - 2r_j k_j z^{-1} + r_j^{-2}} = \frac{2r_j z^{-1}}{D(z)} \qquad (3\text{-}2)$$

where:

$$D(z) = 1 - 2r_j k_j z^{-1} + r_j^{-2} \quad \text{[denominator of } H_{BPj}(z)\text{]}$$

The partial derivative is given by (see equation 2-10):

$$\frac{\partial T(z)}{\partial k_j} = -\prod_{\substack{i=1 \\ i \neq j}}^{N} H_{Ni}(z)\, H_{BPj}(z) H_{sj}(z) \qquad (3\text{-}3)$$

The only parameter that is modified in the adaptation process is the pole cosine k_j for each bandpass filter section j. This parameter also effects the denominator of the sensitivity filter $H_{sj}(z)$. The only other parameter, r_j, is fixed. Selection of r_j will be covered in section 3.3. The standard LMS algorithm would give us the following update procedure for adapting k_j:

$$k_j' = k_j - \mu e_j(z) \frac{\partial T(z)}{\partial k_j} \qquad (3\text{-}4)$$

where: μ is a fixed step size (typically μ=0.1)

As derived in chapter 2, the correct choice for the error function e(z) is:

$$e_j(z) = T(z) \qquad \text{(for all j)} \qquad (3\text{-}5)$$

However, an alternative choice which can be derived by forming the gradient of the section outputs $Y_j(z)$ is:

$$e_j(z) = Y_j(z) \qquad (3\text{-}6)$$

Both of these choices for $e_j(z)$ were tested and found to work well.

A word of explanation is in order here. Equation (3-4) seems to imply that we are mixing z-domain and time domain terms. In a sense, we are. The derivative are taken on the frequency domain

terms, but the outputs labeled $\partial T(z)/\partial k_j$ in Figure 2-4 are actually time series. It is these time series that are used in equation (3-4). Also, the output $e_j(z)$ is really the time series $y_j(i)$. This, of course, implies that we are using instantaneous values to represent the actual values, thus these are approximations to the actual values.

3.2 Forgetting Factor

Use of the instantaneous values as discussed in section 3.1 does not result in a very robust adaptation algorithm. One solution to this problem is to dynamically adjust the step size μ. The justification for this and the mathematical derivation of the "forgetting-factor" concept is well known (eg: see [5]). In order to test various forgetting factors, we employed the following update equation:

```
kj[k]:=kj[k]-step*gradient*(1.0+p[k,0]/pmax)/(pmin+p[k,0])
```

where:

step = μ (fixed for a particular structure) (3-7a)

$$\text{gradient} = e_j(z)\frac{\partial T(z)}{\partial k_j} \qquad (3\text{-}7b)$$

p[k,0] = output of forgetting filter (3-7c)

pmax = limits the effect of large p[k,0] (3-7d)

pmin = limits the effect of small p[k,0] (3-7e)

NOTE:

We found that pmax and pmin were not required so in reality the update algorithm could be replaced by:

```
nj[k]:=kj[k]-step*gradient/p[k,0]                    (3-8)
```

However, all of the data in this chapter was run with equation 7 setting pmax=1000000 and pmin=0.000001.

Three different forgetting filters were tested. The first, called zero-order forgetting, uses the sum of the squares of the instantaneous outputs $\partial T(z)/\partial k_j$ as p[k,0]. The next, called first order forgetting, is a simple single-pole first order low-pass filter applied to the section output $[\partial T(z)/\partial k_j]^2$. The third type, called second-order forgetting, is a second-order low-pass filter applied to $[\partial T(z)/\partial k_j]^2$.

Zero Order Forgetting

$$p[k,0] = \sum \left[\frac{\partial T(z)}{\partial k_j}\right]^2 \qquad (3\text{-}9a)$$

First Order Forgetting

$$p[k,0] = (1-r_f)\left[\frac{\partial T(z)}{\partial k_k}\right]^2 + r_f p[k,0] \qquad (3\text{-}9b)$$

Second Order Forgetting

p[k,0] = output of one of the following second-order filters with input $[\partial T(z)/\partial k_j]^2$:

Simple 2nd-Order Low-Pass

$$H_1(z) = \frac{h_1(1 - 2z^{-1} + z^{-2})}{1 - 2r_f k_j z^{-1} + z^{-2}} \qquad (3\text{-}9c)$$

Complex 2nd-Order Low-Pass

$$H_2(z) = \frac{h_2(1 - 2\cos(5\theta_j)z^{-1} + z^{-2})}{1 - 2r_f k_j z^{-1} + z^{-2}} \qquad (3\text{-}9d)$$

where: $\theta_j = \text{ArcCos}(k_j)$

The filter of equation (3-9d) was used by Kwan and Martin [5].

3.3 Selection of Notch Bandwidth

Each bandpass filter has a fixed bandwidth set by the parameter r_j. If we let x be the lower 3db frequency of the notch and y be the upper 3db frequency of the notch, then the bandwidth is d = x - y. We shall first calculate d at the frequency z=j (90°, k_j=0). Then we shall calculate x and y for other values of k_j based on the fixed d.

When k_j=0 we can solve for the two 3db frequencies x_0 and y_0 from the following equation:

$$\frac{1}{2} = |H_N(z)|^2 = \left[\frac{1+r^2}{2}\right]^2\left[\frac{(1+\cos 2\omega t)^2 + \sin^2 2\omega t}{(1+r^2\cos 2\omega t)^2 + r^4\sin^2 2\omega t}\right] \qquad (3\text{-}10a)$$

$$1 = \left[\frac{1 + 2r^2 + r^4}{4}\right]\left[\frac{4 + 4\cos 2\omega t}{(1 + 2r^2\cos 2\omega t + r^4}\right] \qquad (3\text{-}10b)$$

cross-multiplying yields:

$$1+2r^2\cos 2\omega t+r^4 = 1+2r^2+r^4 + \cos 2\omega t(1+2r^2+r^4) \qquad (3\text{-}11)$$

Using the fact that $\cos 2\omega t = 2\cos^2\omega t - 1$, we solve equation (3-11) for $\cos\omega t$:

$$\cos 2\omega t = 2\cos^2\omega t - 1 = \frac{-2r^2}{1+r^4} \qquad (3\text{-}12a)$$

$$\cos\omega t = \pm \frac{1-r^2}{\sqrt{2(1+r^4)}} \qquad (3\text{-}12b)$$

Since x_0 is the lower 3db frequency and y_0 is the upper 3db frequency, we have:

$$x_0 = \text{ArcCos}\left[\frac{-(1-r^2)}{\sqrt{2(1+r^4)}}\right] \qquad (3\text{-}13a)$$

$$y_0 = \text{ArcCos}\left[\frac{1-r^2}{\sqrt{2(1+r^4)}}\right] \qquad (3\text{-}13b)$$

Also, since $\sin^2\theta + \cos^2\theta = 1$, we have:

$$\cos(x_0) = -\frac{1-r^2}{\sqrt{2(1+r^4)}} \qquad (3\text{-}14a)$$

$$\cos(y_\theta) = + \frac{1-r^2}{\sqrt{2(1+r^4)}} \qquad (3\text{-}14b)$$

$$\sin(x_\theta) = + \frac{1+r^2}{\sqrt{2(1+r^4)}} \qquad (3\text{-}14c)$$

$$\sin(y_\theta) = + \frac{1+r^2}{\sqrt{2(1+r^4)}} \qquad (3\text{-}14d)$$

We may how calculate the bandwidth d from $d = x_\theta - y_\theta$:

$$\cos(d) = \cos(x_\theta - y_\theta)$$

$$= \cos(x_\theta)\cos(y_\theta) + \sin(x_\theta)\sin(y_\theta) \qquad (3\text{-}15)$$

Substitution from equations (3-14) into equation (3-15) yields:

$$\cos(d) = \frac{-(1-2r^2+r^4) + (1+2r^2+r^4)}{2(1+r^4)} = \frac{2r^2}{1+r^4} \qquad (3\text{-}16)$$

Although we have calculated d for the specific case of a notch at $z=j$ (90°), d is constant at all frequencies. In general the notch frequency $z=e^{jf}=\cos(fT)+j\sin(fT)$ for any section can be calculated as follows:

$$\left[\frac{1+r^2}{2}\right] - 2rk_j z^{-1} + \left[\frac{1+r^2}{2}\right] z^{-2} = 0 \qquad (3\text{-}17a)$$

$$\left[\frac{1+r^2}{2}\right]\left[1 - \frac{4rk_j}{1+r^2} z^{-1} + z^{-2}\right] = 0 \qquad (3\text{-}17b)$$

Substituting for $z=e^{jf}=\cos(f\tau)+j\sin(f\tau)$ in equation (3-17b) and setting the real and the imaginary part to zero yields:

$$1 - \frac{4rk_j}{1+r^2}\cos(f\tau) + \cos(2f\tau) = 0 \qquad (3\text{-}18a)$$

$$\frac{4rk_j}{1+r^2}\sin(f\tau) + \sin(2f\tau) = 0 \qquad (3\text{-}18b)$$

Using trigonometric identities:

$$1 - \frac{4rk_j}{1+r^2}\cos(f\tau) + 2\cos^2(f\tau) - 1 = 0 \qquad (3\text{-}19a)$$

$$\frac{4rk_j}{1+r^2}\sin(f\tau) + 2\sin(f\tau)\cos(f\tau) = 0 \qquad (3\text{-}19b)$$

both equations yield the same result:

$$\cos(f\tau) = \frac{2rk_j}{1+r^2} \qquad (3\text{-}20)$$

Note that the notch frequency is only slightly different from the pole frequency determined by $k_j=\cos(\theta)$. If r=1, the two frequencies would be identical. Of course, for stability, r must be less than one. Table 3-I lists the relationship of d to the value of r. From the table we see that values of r around .9 are well suited to the type of notches needed to eliminate narrow-band interference. The table also lists the measured values for the actual notch frequency *f*, the lower 3db frequency x, and the upper 3db frequency y for each of the notches. These compare closely to the calculated values in equation (3-16), equation (3-20), and as follows:

Table 3-I Effect of r on Notch Width

r	θ	x	*f*	y	d
0.300	0.083	0.093	0.171	0.314	0.221
0.300	0.167	0.123	0.206	0.344	0.221
0.300	0.250	0.163	0.250	0.384	0.221
0.500	0.083	0.068	0.128	0.240	0.172
0.500	0.167	0.118	0.185	0.290	0.172
0.500	0.250	0.178	0.250	0.350	0.172
0.700	0.083	0.059	0.099	0.164	0.105
0.700	0.167	0.127	0.172	0.232	0.105
0.700	0.250	0.203	0.250	0.308	0.105
0.800	0.083	0.062	0.090	0.131	0.069
0.800	0.167	0.138	0.169	0.207	0.069
0.800	0.250	0.218	0.250	0.287	0.069
0.900	0.083	0.070	0.085	0.103	0.033
0.900	0.167	0.151	0.167	0.185	0.033
0.900	0.250	0.234	0.250	0.267	0.033
0.950	0.083	0.076	0.084	0.092	0.016
0.950	0.167	0.159	0.167	0.175	0.016
0.950	0.250	0.242	0.250	0.258	0.016

$$y - x = d \tag{3-21a}$$

$$xy = f^2 \tag{3-21b}$$

hence:

$$y^2 - xy - dy = y^2 - f^2 - dy = 0 \tag{3-22a}$$

$$y = d/2 + \sqrt{(d/2)^2 + f^2} \qquad x = d/2 - \sqrt{(d/2)^2 + f^2} \tag{3-22b}$$

3.4 Minimum and Maximum Frequency of Notch

Equation (3-20) implies that the notch frequency is slightly different from the pole frequency θ. In particular:

$$\cos(f_\tau) = \frac{2rk_j}{1+r^2} = \frac{2r}{1+r^2}\cos(\theta) \tag{3-20}$$

This difference becomes smaller as r approaches one. However, if the pole frequency θ is at its minimum of 0°, then $k_j=1$ and:

$$\Delta = f_\tau\Big|_{\theta=0} = \text{ArcCos}\left[\frac{2r}{1+r^2}\right] \tag{3-21a}$$

If θ is at its maximum of π (180°), then $k_j=-1$ and:

$$\pi - \Delta = f_\tau\Big|_{\theta=\pi} = \text{ArcCos}\left[\frac{-2r}{1+r^2}\right] \tag{3-21b}$$

From equation (3-20) we see that the difference between the notch frequency f_τ and the pole frequency θ is at most Δ. As θ approaches $\pi/2$ (90°), the difference between the notch and pole

frequencies goes to zero. Table 3-II gives Δ and 180°-Δ for various values of r.

Table 3-II Effect of r on Δ

r	Minimum Notch frequency Δ	Maximum Notch frequency 180°-Δ
0.3	56.60151°	123.39849°
0.5	36.86990°	143.13010°
0.7	20.01596°	159.98404°
0.8	12.68038°	167.31962°
0.9	6.02558°	173.97442°
0.95	2.93760°	177.06240°

The maximum and minimum notch frequencies given in Table 3-II are important because if there is a sinusoid at a frequency lower than the minimum notch frequency or if there is a DC value, the notch will be forced to this minimum frequency. Similarly, if there is a sinusoid above the maximum frequency or at the Nyquist frequency, the notch will be forced to this maximum frequency. Figure 3-1 shows the result of a DC signal on the adaptation process. The notch is forced to 6.02558° in this case because the radius of the notch filter is chosen as r=0.9. Similarly, Figure 3-2 shows a signal at the Nyquist frequency forcing the notch to the maximum frequency of 173.99442°.

3.5 Algorithm Performance

Critical to the performance of the algorithm is the choice of the various parameters:

1. Number of notch filters,
2. Notch width r_j,
3. Zero, first, or second-order forgetting (or none),
4. Step size μ,
5. Limit on the parameter k, and
6. Limit on the parameter p.

The first three choices are somewhat independent of each other. But the choice of the step size μ is highly dependent on the first three choices. We found that limiting k such that $-1 \leq k_j \leq 1$ was very important in all cases. However, we found that limits on p were not very useful.

The choice of the number of notch filters has a profound effect on the algorithm performance, but is assumed to be out of our control. We assume that the number of notches is fixed at some number by the application. Thus in evaluating the algorithm, we look at three cases:

Case 1: Performance with only one notch,

Case 2: Performance with five notches,

Case 3: Performance with ten notches.

Table 3-III summarizes experiments with the new algorithm. For these experiments, the following 10 frequencies were used:

1. $f_{in} = 0.08333\ f_s$ (30°)
2. $f_{in} = 0.1\ f_s$ (36°)
3. $f_{in} = 0.125\ f_s$ (45°)
4. $f_{in} = 0.1667\ f_s$ (60°)
5. $f_{in} = 0.2\ f_s$ (72°)
6. $f_{in} = 0.25\ f_s$ (90°)
7. $f_{in} = 0.3\ f_s$ (108°)
8. $f_{in} = 0.3333\ f_s$ (120°)
9. $f_{in} = 0.375\ f_s$ (135°)
10. $f_{in} = 0.4\ f_s$ (144°)

In cases where only one frequency was selected, the results are based on the results with each of the 5 frequencies (1, 3, 6, 8, 10) taken separately. In cases where three frequencies were used, the following five combinations were used {(1,2,6),

(1,3,10), (1,6,10), (1,9,10), (6,9,10)}. When 5 frequencies are selected, we used the 6 combinations of the following 6 frequencies (1, 3, 4, 6, 8, and 9) taken 5 at a time.

3.5 Conclusions

Table 3-IV shows the results of these experiments. From these results, we conclude that a forgetting factor is needed for multi-notch filters and that a second-order forgetting factor does not seem to inprove the performance over a first-order factor which seems to work quite satisfactorilly. The first-order forgetting-factor is the only one that does not exhibit "jump" convergence. Here we refer to "jump" convergence as the situation when there are more interferers than notches and the last notch in the cascade "jumps" between the remaining frequencies to be eliminated. This "jump" convergence, might actually be desireable in some instances, in which case one would want to avoid the first-order forgetting factor.

A word should be given about the zero-order forgetting. For this case alone, the parameters p_{min} and p_{max} were not set to the default values of p_{min}=0.000001 and p_{max}=1000000. For zero-order forgetting, it is necessary to have specific values for p_{min} and p_{max} in order to obtain convergence. The "Limit on p value" column in the table gives the required limits to obtain convergence for the zero-order forgetting factor. It should be kept in mind that this column applies ONLY to the zero-order forgetting factor data. All other data was obtained with the default values for p_{min} and p_{max}. It also should be noted that limits on p_{min} and p_{max} for the other forgetting factors can improve performance, but at the cost of a more comples update algorithm.

Table 3-IV Summary of Tests on New Algorithm

Case	# of 2-nd Order secs	Sines	Limit on p value		Noise	No Forgetting		Zero-Order Forgetting		1st-Order Forgetting		2nd-Order Forgetting	
			p_{min}	p_{max}		μ	iter	μ	iter	μ	iter	μ	iter
1a	1	1	0.05	20	NO	.02	50	.05	67	.05	20	.05	20
1an	1	1	0.05	20	YES	.02	50	.05	Jump	.05	20	.05	144
1b	1	3	0.05	20	NO	.02	700	.05	Jump	.05	30	.05	25
1bn	1	3	0.05	20	YES	.02	750	.05	Jump	.05	50	.05	100
2a	5	3	0.10	10	NO	1.46	2500	.017	340	.063	200	.07	400
2an	5	3	0.10	10	YES	1.46	150	.017	1200	.063	400	.07	450
2b	5	5	0.10	10	NO	1.46	325	.017	1410	.063	150	.07	500
2bn	5	5	0.10	10	YES	1.46	3120	.017	3500	.063	175	.07	250
2c	5	7	0.10	10	NO	1.46	Jump	.017	1667	.063	450	.07	Jump
2cn	5	7	0.10	10	YES	1.46	Jump	.017	700	.063	167	.07	Jump
3a	10	7	0.02	50	NO	1.46	2000	.02	1667	.075	500	.07	1500
3an	10	7	0.02	50	YES	1.46	9500	.02	5000	.075	750	.07	1000
3b	10	10	0.02	50	NO	1.46	7500	.02	7500	.075	2000	.07	5250
3bn	10	10	0.02	50	YES	1.46	1000	.02	1000	.075	2000	.07	2500

1. In all cases without noise, parameters converge to exact value when number of notches is greater or equal to number of sines. Iterations listed in table represent convergence to three decimal places of correct value.

2. In all cases with noise, parameters oscillate around exact value. The number of iterations listed in the table is the number of iterations required to establish this oscillatory pattern.

3. In some cases with more interfering sine waves than notches, the notches will jump at random between sine waves. This pattern is indicated in the table by the word "Jump".

4. Values of p_{min} and p_{max} in the table refer only to zero-order forgetting which generally will not converge without limits on p. All other types of forgetting were run without limits on p (ie: $p_{min}=10^{-6}$ and $p_{max}=10^{+6}$).

4.0 Algorithm Performance and Simulation

The adaptive digital filter algorithm as described in the earlier chapters is simulated and tested using synthetic data. Simulation was carried out on VAX 11/785 computer using Fortran 77. The synthetic data generated for testing this algorithm is of four categories:

1. Sinusoidal signals with white gaussian noise
2. Narrow Band Noise with white gaussian noise
3. Bi-Phase Shift Keying (BPSK) sequence
4. Frequency Shift Keying (FSK) sequence

4.1 Sinusoidal signals

For generating sinusoidal signals placed at different normalized frequencies θ, a second order AR process given as:

$$x^i_k = 2\cos(\theta_i)x^i_{k-1} - x^i_{k-2} \qquad (4\text{-}1)$$

was used with poles on the unit circle. This approach was chosen in order to reduce the computational burden. Using equation 4-1, we find that initial conditions are very important and they are chosen such that $x_{-1}=0$ and $x_{-2}=-\sin(\theta_i)$ giving a unit amplitude sinusoidal signal. The θ_i value is between 0 to 180 and n is the number of frequencies desired. The required signal y_k needed to input into the adaptive algorithm is given as:

$$y_k = \sum_{i=1}^{i=n} x^i_k + \gamma_k \qquad (4\text{-}2)$$

where γ_k is a white gaussian noise $N(0,\sigma^2)$.

4.2 Narrow Band Noise

The narrow band signal is generated using the difference equation:

$$z^i_k = 2r\cos(\theta_i)z^i_{k-1} - r^2x^i_{k-2} + u^i_k \qquad (4\text{-}3)$$

where θ_i decides the placement of the noise in the spectrum and r controls the band-width of the noise. The u^i_k is simply a uniformly distributed noise taken at different instants. The desired signal y_k is obtained via:

$$y_k = \sum_{i=1}^{i=n} z^i_k + \gamma_k \qquad (4\text{-}4)$$

4.3 Bi-Phase Shift Keying Sequence

The generation of BPSK signal has distinctly three parts which are generated as shown in Figure 4-1. The three parts are as follows:

1. Generation of the Random Binary Sequence(RBS) is achieved by passing a uniformly distributed noise through a hard limiter. (An important note is that the interval between the two consecutive bits of RBS is $1/f_b$.)

2. Generation of another sequence of binary numbers which are in practice a *spreading code* or *spreading sequence*. The specific sequence used in a given communications system is normally not available to anyone but the intended receiver. (In this particular simulation we have generated the spreading sequence by passing a uniformly distributed noise through a hard limiter. The bit interval is $1/f_c$.)

3. Generation of the phase encoding, ie: the mapping of the given binary signal which is the *exclusive or* of (1) and (2) above into either $\emptyset$ or π at *appropriate* sample times.

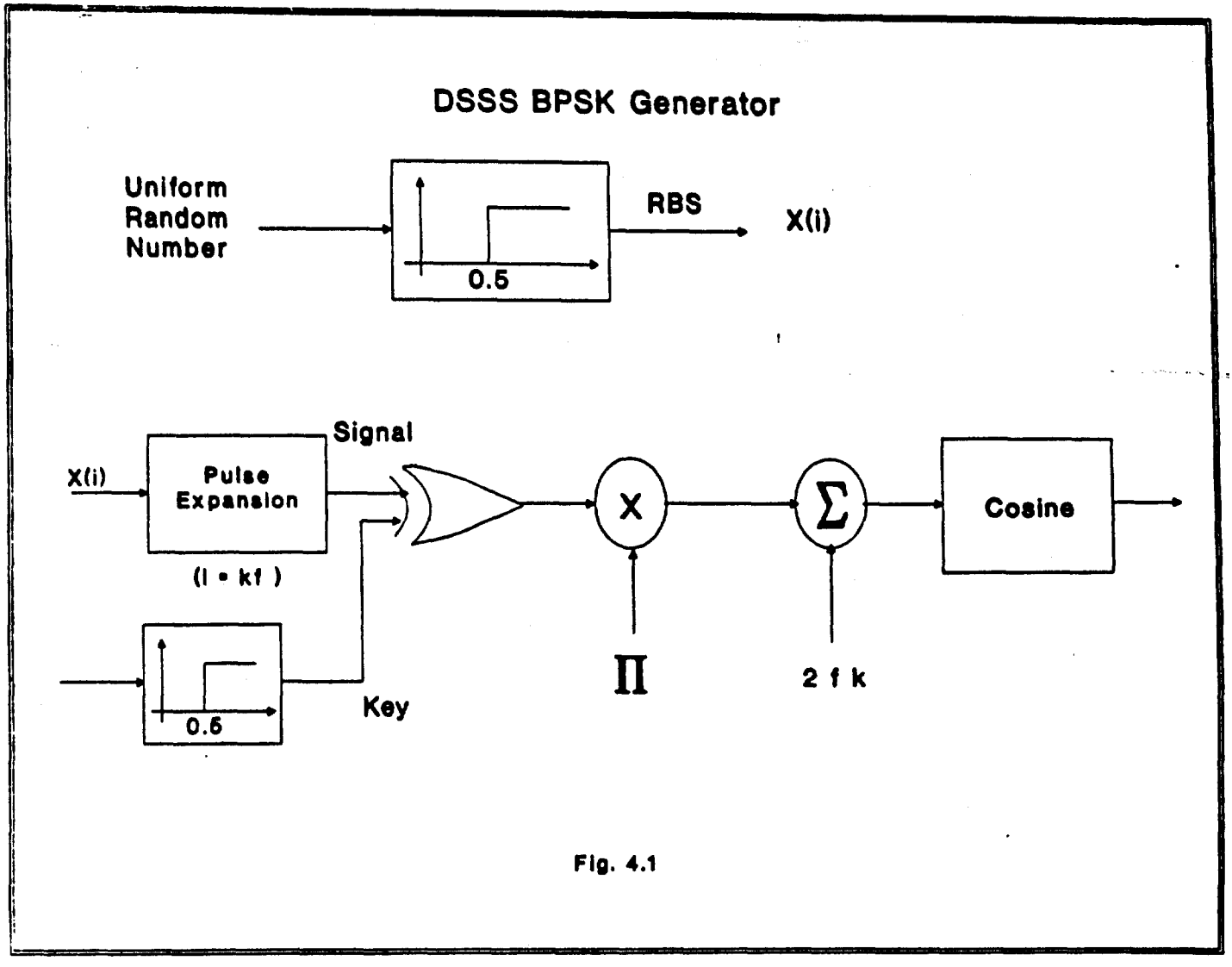


Fig. 4.1

Figure 4-1. Generation of the BPSK Signal.

The output of the first hard limiter is stored in an array x (see Figure 4-1). Output of the second hard limiter is stored in the array y. This information is retrieved by a subtle use of the array index given as $i=kf_b$ where i is the index, k is the discrete sample number and f_b is the bit rate of the intelligence. Similarly another index j is generated using $j=kf_c$ where f_c is the chip frequency Generation of the indices is the key trick in this simulation. The desired signal now is given by the equations:

$$w^P_k = A\cos(2\pi f_0 k + \Phi_k) \qquad (4\text{-}5)$$

$$\Phi_k = [x(i) \oplus y(j)]\pi \qquad (4\text{-}6)$$

$$\alpha_k = \sum_{P=1}^{P=n} \omega^P_k \qquad (4\text{-}7)$$

where i and j are indices of the arrays as defined earlier. It is an important point to note that the signal ω^P_k is completely defined by the parameter $p = \{f_0, f_c, f_b\}$.

This signal is not really a simple BPSK signal but it has an additional feature of spreading the spectrum by controlling the chip-frequency and carrier frequency and baud-rate. The block diagram of the scheme is given in Figure 4-1. The desired signal y_k is given by:

$$y_k = \alpha_k + \beta_k \qquad (4\text{-}8)$$

Where: α_k is the narrow band BPSK signal placed at different parts of the frequency spectrum and β_k is the broad band BPSK signal generated for a specific p value.

4.4 Frequency Shift Keying Sequence

Generating this sequence needs a random binary intelligence signal. This was once again achieved by passing a uniformly distributed noise through a hard limiter. The output of the same stored in an array x. An index i is chosen such that $i=kf_b$ where f_b is the baud-rate of the information and k is discrete sample number. Now the desired signal is generated via:

$$s_k = 2\cos(\theta_k)s_{k-1} - s_{k-2} \qquad (4\text{-}9)$$

$$\theta_k = \theta + \delta x(i) \qquad (4\text{-}10)$$

$$y_k = s_k + \gamma_k \qquad (4\text{-}11)$$

where θ is the carrier frequency and δ is the depth of the frequency modulation. Initial conditions are very important and they are chosen such that $s_{-1}=0$ and $s_{-2}=-\sin(\theta)$ giving an unit amplitude sinusoidal signal.

4.5 Simulation

The adaptive filter algorithm as described in the previous chapters will now be simulated and tested using synthetic data generated as described in the last sections. These simulations were carried out on a VAX 11/785 computer and were written in Fortran 77. Further details on these simulations can be found in reference [12]. The adaptive filters used are the Kwan and Martin filter and the modified Kwan and Martin filter (new algorithm) as described in chapters 2 and 3. The key parameters of these filters are:

(a) Sharpness of the notch filter defined by pole position ($r^2 = 1-k_2$)
(b) Step size in the incrementation procedure (μ)
(c) Time constant of the fading filter (λ)
(d) Model order (n)
(e) Order of incoming signal or number of interferers (m)

4.5.1 Fading Filter

In order to track the time-varying parameters it is essential to compute $(1/N)\sum_i e^2_k$ over some period of time and this can be essentially achieved by a simple averaging filter:

$$v_k = \lambda v_{k-1} + (1 - \lambda)e^2 \qquad (4\text{-}12)$$

with a time constant λ and this is mostly referred to as the *forgetting factor* or *fading factor*. (NOTE: This is equivalent to the first-order forgetting of equation 3-9b.)

The algorithm was fine tuned for the factors (b) and (c) by using pure sinusoidal signal. This was achieved by manually optimizing μ and λ. The value of μ was chosen as 0.01 and λ=0.9.

After fixing the values of μ and λ the algorithm was tested using Narrow Band Noise signal y_k for its performance. This signal was used only to tune the value of r (i.e. sharpness of the notch filter). It was observed that if the value of r is close unity variance of the parameter k^i_1 estimate increases. Based on simulation the value of r was chosen as 0.9.

4.5.2 Check VAX version against IBM-PC version

Under these parameter conditions BPSK signal y_k was applied to the adaptive algorithm. Figure 4-2 shows the block diagram of the new filter that will be tested in this chapter. Other than notational changes, this is the same as Figure 2-4. Figure 4-3 shows the results of a sine-wave test like was described in chapter 3. The purpose of this test was to verify that the VAX version of the program gave the same results as the IBM-PC version used in the previous sections. Figure 4-3a was the input consisting of three sine waves in the presence of noise, but no BPSK signals. Figure 4-3b is the filtered output, demonstrating that the filter did remove the sine waves. Figure 4-3c shows the parameter adaptation. By making a detailed comparison of this example on both the VAX and the IBM-PC, we were able to verify that the VAX program was working the same as the IBM-PC program.

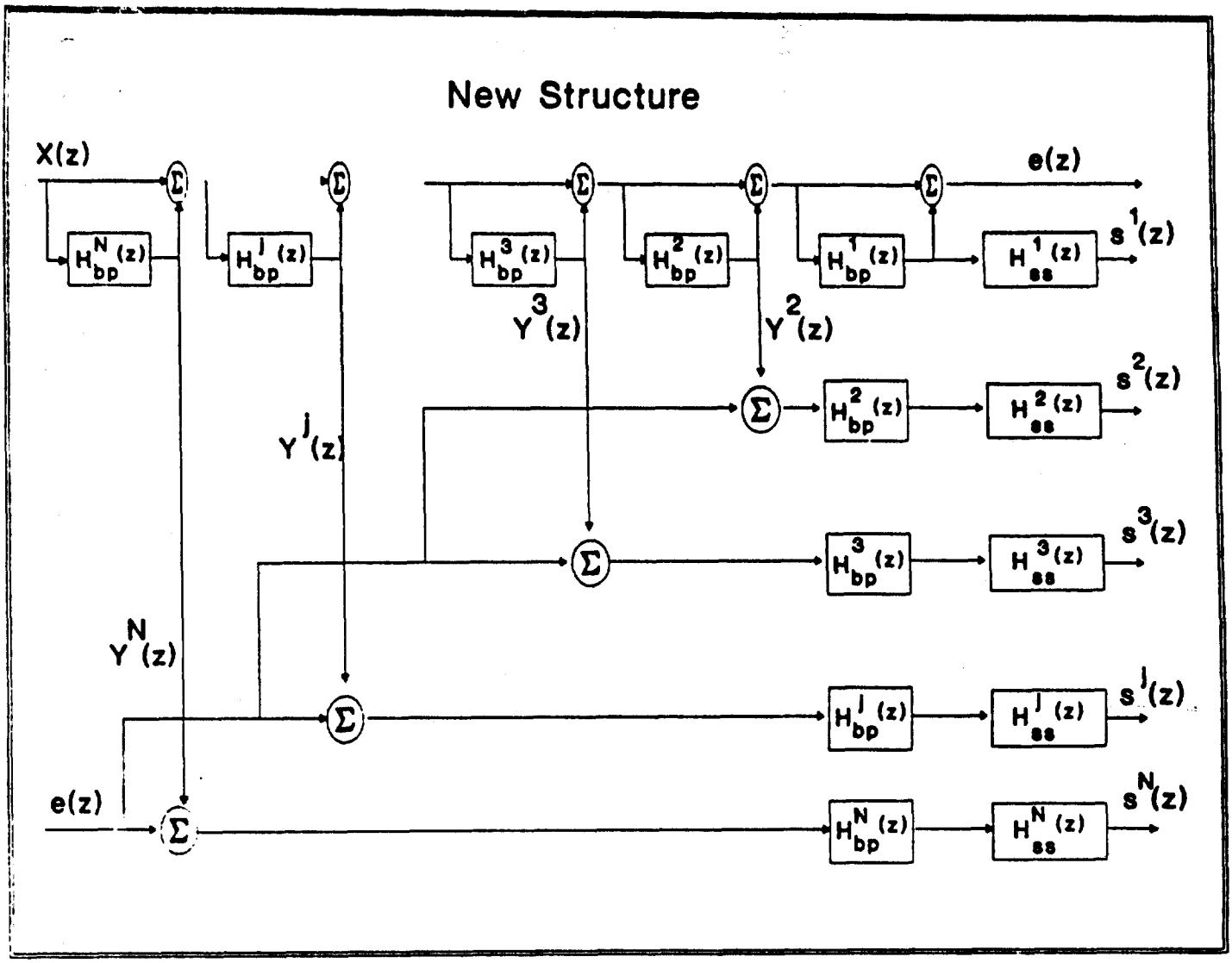


Figure 4-2. Block Diagram of New Algorithm.
(NOTE: Same as Figure 2-4, except for notation)

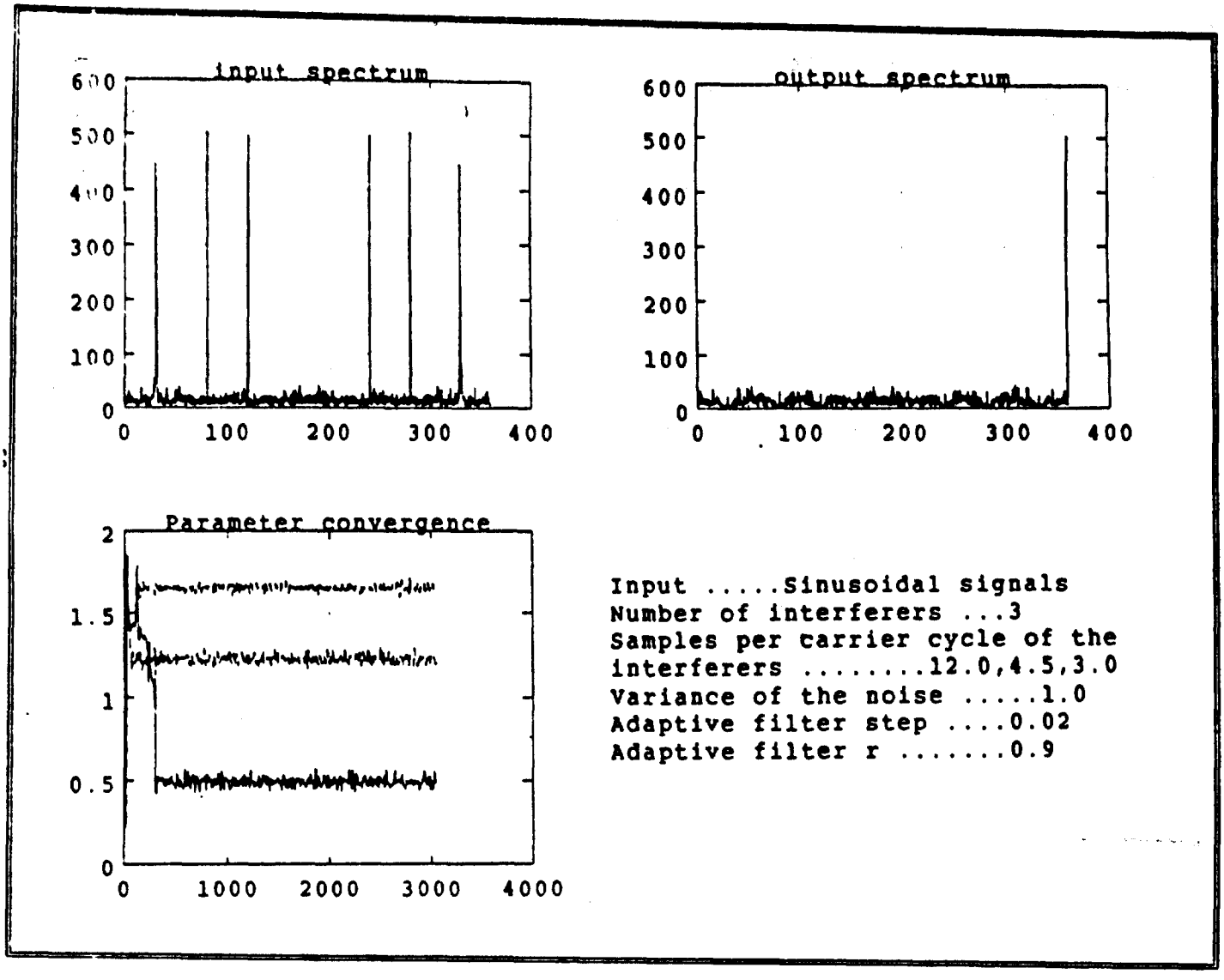


Figure 4-3. Three Pure Sinusoids in Gausian Noise

4.5.3 Elimination of narrow-band BPSK interference

Our next experiment was designed to see how the new algorithm performed on narrow-band BPSK signals rather than sine wave interferers. Figure 4-4a shows the input consisting of three narrow-band BPSK signals at the same frequencies as the sine waves in Figure 4-3a. Figure 4-4b shows the filtered output verifying that the new filter is able to eliminate narrow-band signals as well as sine waves. Figure 4-4c shows the parameter adaptation for this case and should be compared to Figure 4-3c.

4.5.4 Transient behavior (tracking a moving interferer)

The transient behavior of the adaptive filter was tested by generating an FSK signal using the hardware of Figure 4-1. The FSK signal is similar to the BPSK signal, but the spectrum is constantly changing. In our example, the spectrum switched back and forth between two frequencies. Figure 4-5a shows the FFT of the input signal. From Figure 4-5a we can see the two distinct frequencies that the input is changing between. Figure 4-5b shows the output spectrum which has been essentially cleaned of the interfering signal. Figure 4-5c shows the parameter adaptation moving between two values. When compared to the input time sequence, we were able to verify that the algorithm was tracking the FSK signal and eliminating it extremely well [10].

4.5.5 Elimination of narrow-band BPSK from broad-band BPSK

Our next experiment was aimed at eliminating narrow-band BPSK interference from a broad-band BPSK signal. Figure 4-6a shows the input consisting of three narrow-band BPSK signals interfering with a broad-band BPSK signal in the presence of white Gaussian noise. All of the signals have the same energy content. Figure 4-6b shows the output spectrum. Comparing this with the desired BPSK signal, we were able to verify that the new algorithm eliminated the noise without adversely effecting the signal [10]. Furthermore, we processed this signal using the Cyclic Spectral Analysis Software Package [13] and obtained the output shown in Figure 4-7. This output was as expected and confirmed that the new adaptive notch filters were performing well [12].

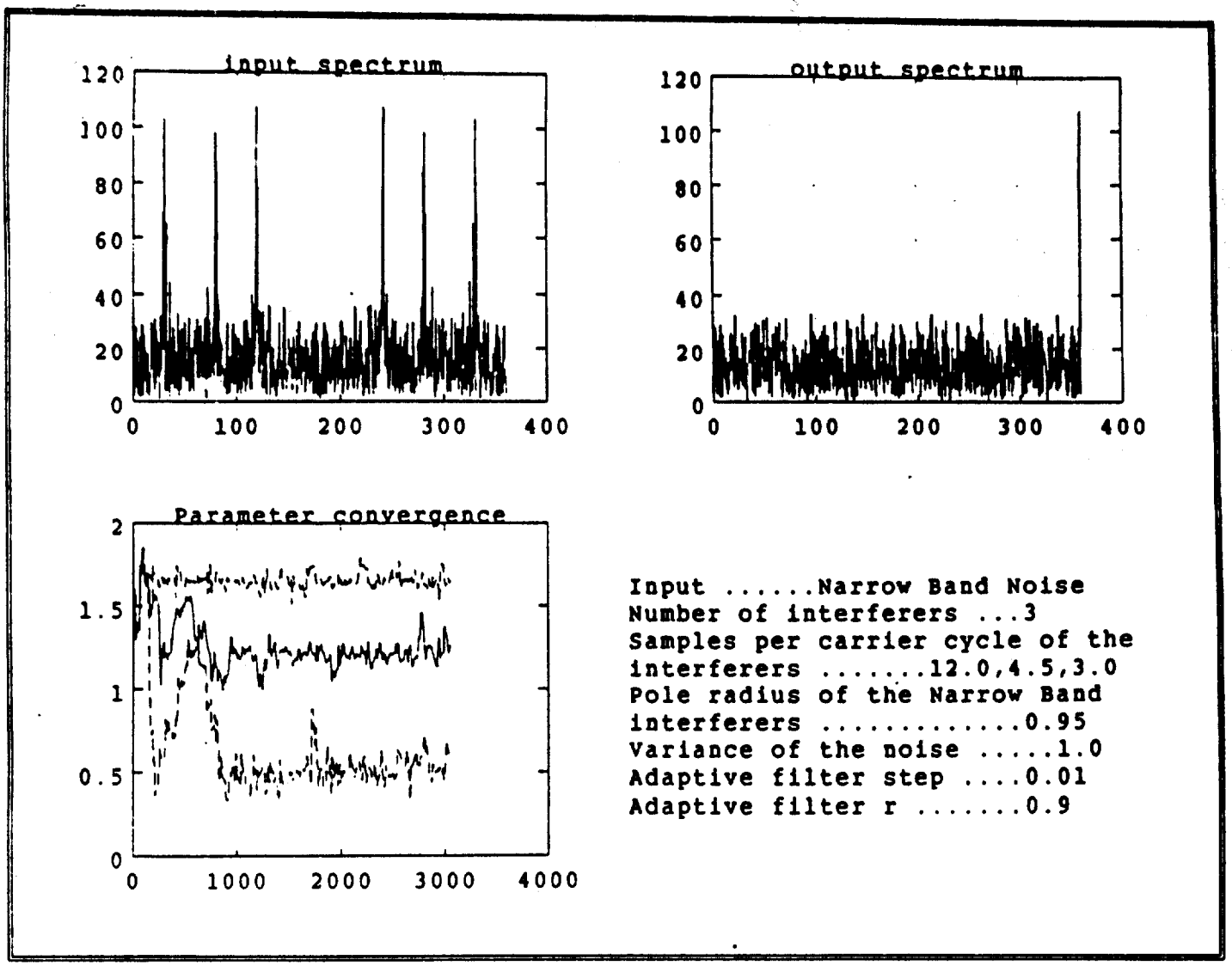


Figure 4-4. Spectrum of 3 Narrow-Band BPSK signals in Noise

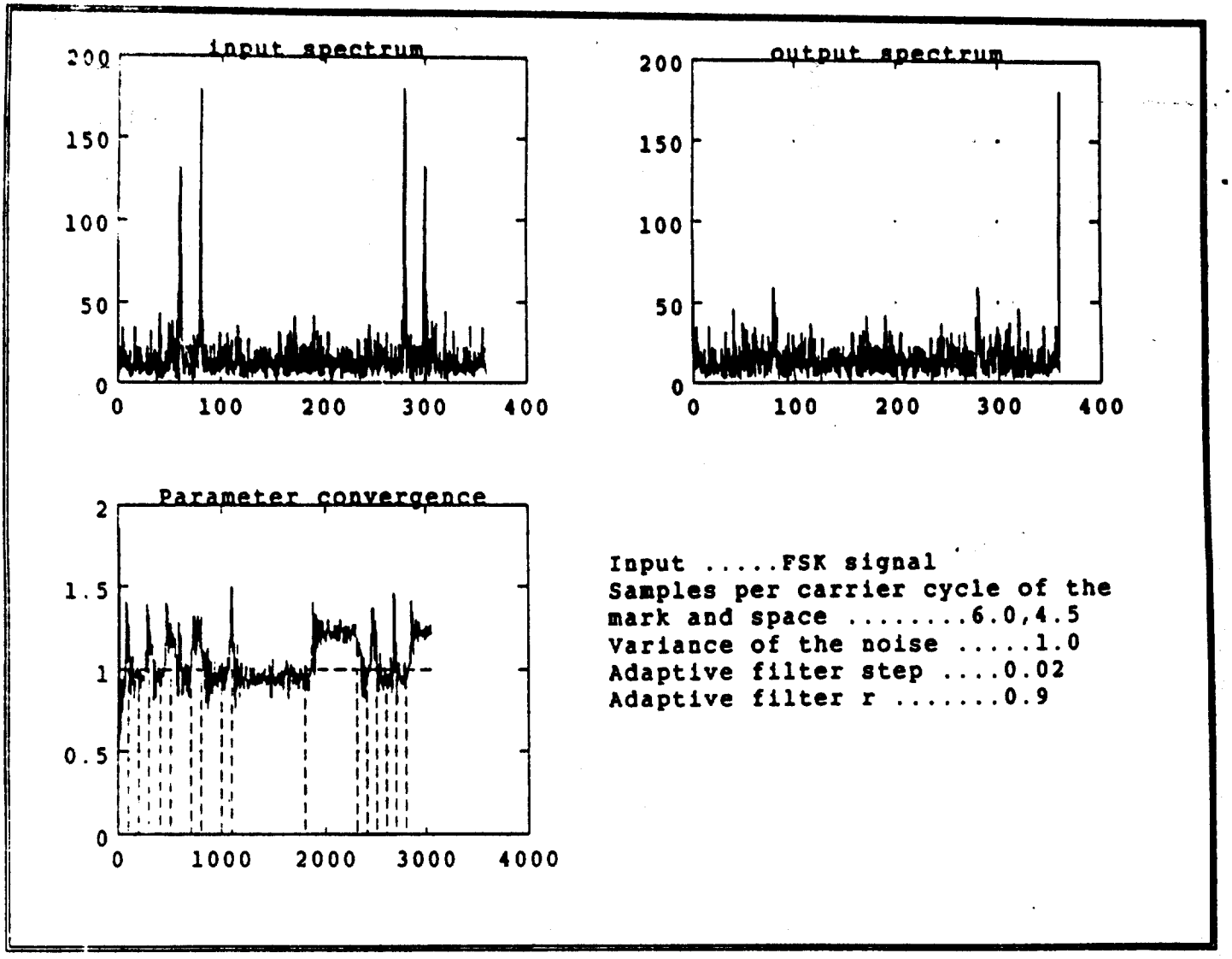


Figure 4-5. Tracking of an FSK Signal in Noise.

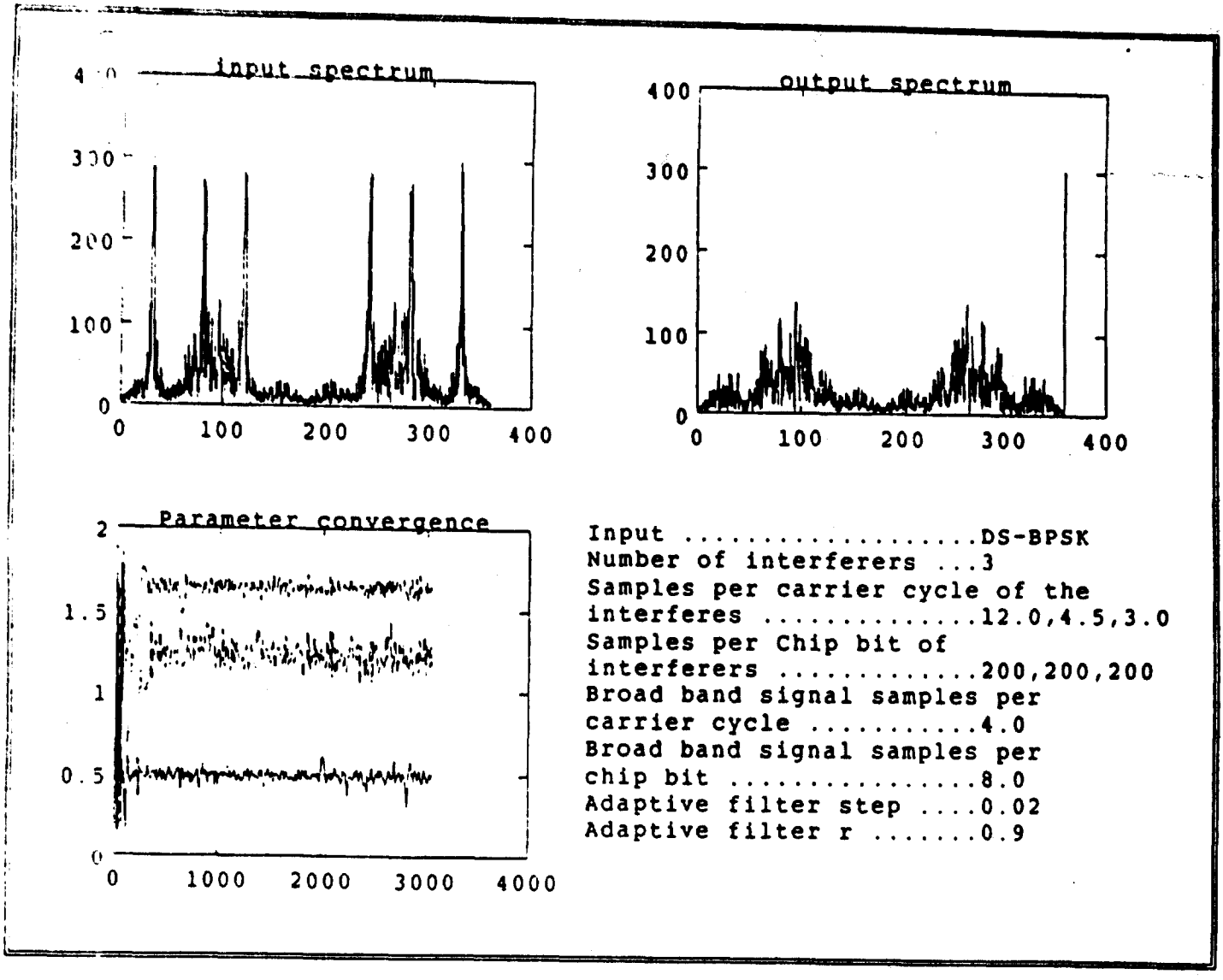
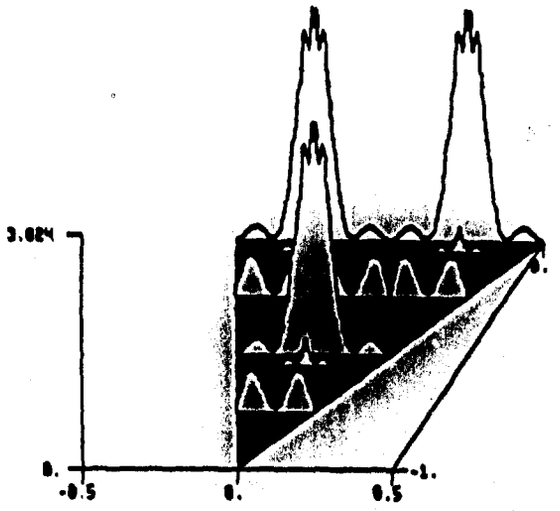
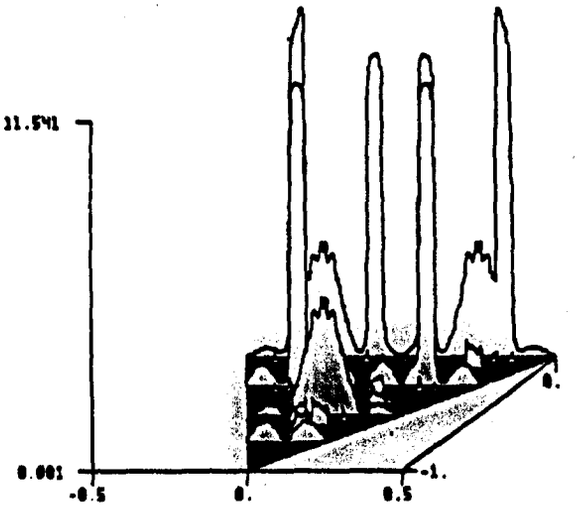


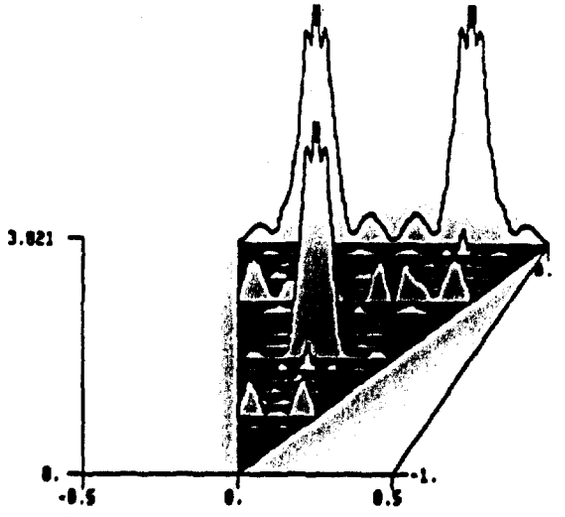
Figure 4-6. Filtering Narrow-Band BPSK from Broad-Band BPSK signal in white Gausian Noise



a. Broad-Band BPSK without Interference or Noise



b. Broad-Band BPSK Corrupted with Interference and Noise



c. Corrupted Output After Processing by New Notch Filters

Figure 4-7. Processing with Cyclic Spectral Analysis Software

4.5.6 Performance with too many notches

One situation of great concern was what would happen if there were too many notches for the number of interfering signals. Would the remaining notches attempt to eliminate the broad-band BPSK signal, and if so would this effect the Cyclic Spectral Analysis Software Package. Figures 4-8 and 4-9 address this important issue. In Figure 4-8 there are two interferers and three notches and in Figure 4-9 there is only one interferer and three notches. As before, Figure 4-8a shows the input spectrum, Figure 4-8b shows the output spectrum, and Figure 4-8c shows the parameter adaptation. As can be seen in Figure 4-8c, the remaining notch wanders around the frequencies of the main lobe of the broad-band BPSK signal attempting to cancel it. However, since the notch is very narrow compared to the main lobe of the broad-band BPSK signal, the notch is unable to eliminate any significant amount of signal energy. Thus when we applied the Cyclic Spectral Analysis Software Package, we got the same results as Figure 4-7 [10]. A similar result was found in Figure 4-9. The two remaining notches in Figure 4-9c wander around the main lobe of the broad-band BPSK signal, but are unable to eliminate any significant amount of signal energy. Thus when the Cyclic Spectral Analysis Software Package was applied, we once again got the results of Figure 4-7 [10].

4.6 Conclusions

The BPSK signal y_k was given as input to the new algorithm and the results showed that the new adaptive notch filters were able to perform well under various situations expected in the real world. The new algorithm consistently fared well in its mission and was able to prepare the incoming data for processing by the Cyclic Spectral Analysis Software Package.

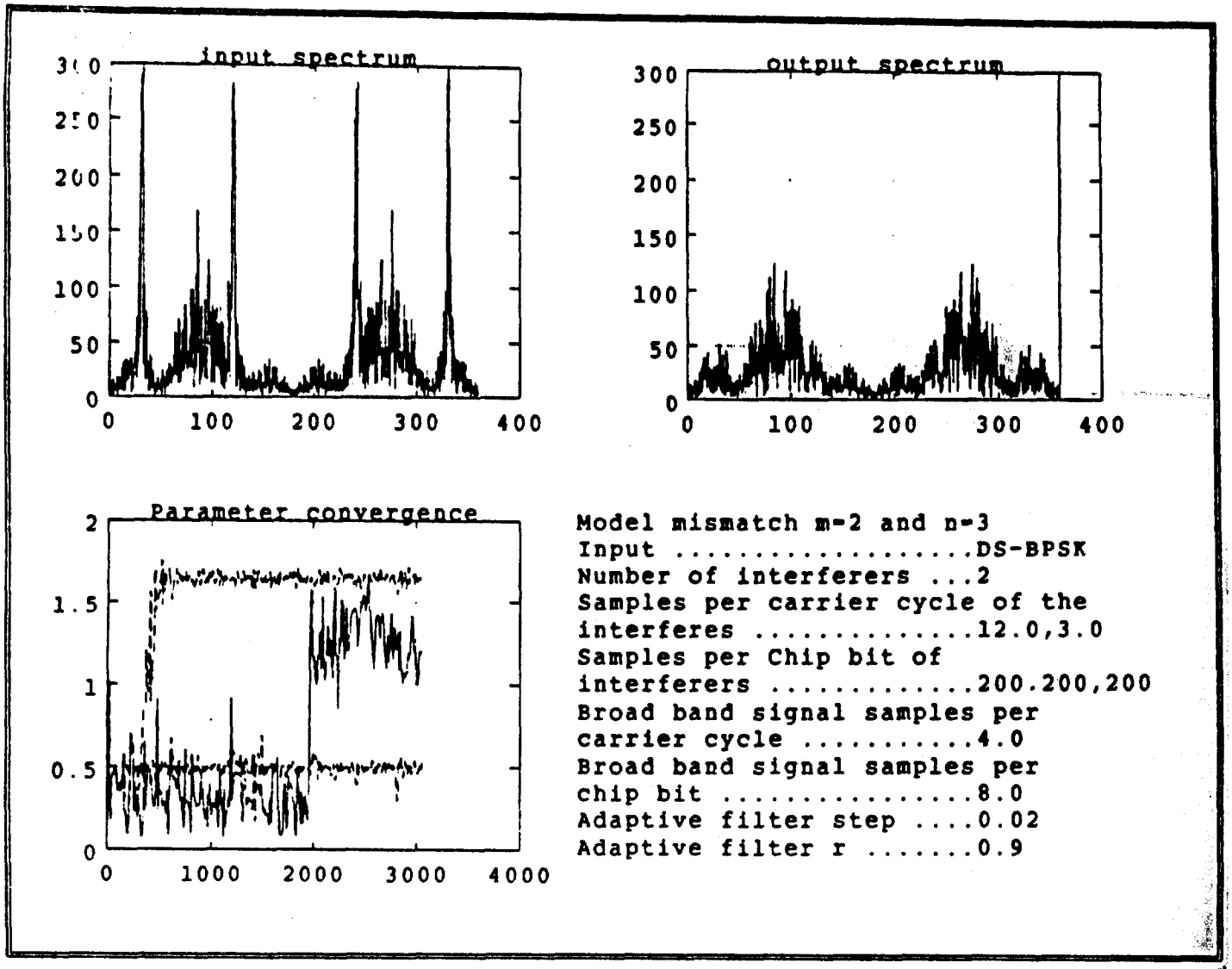


Figure 4-8. Two Interferers and Three Notches

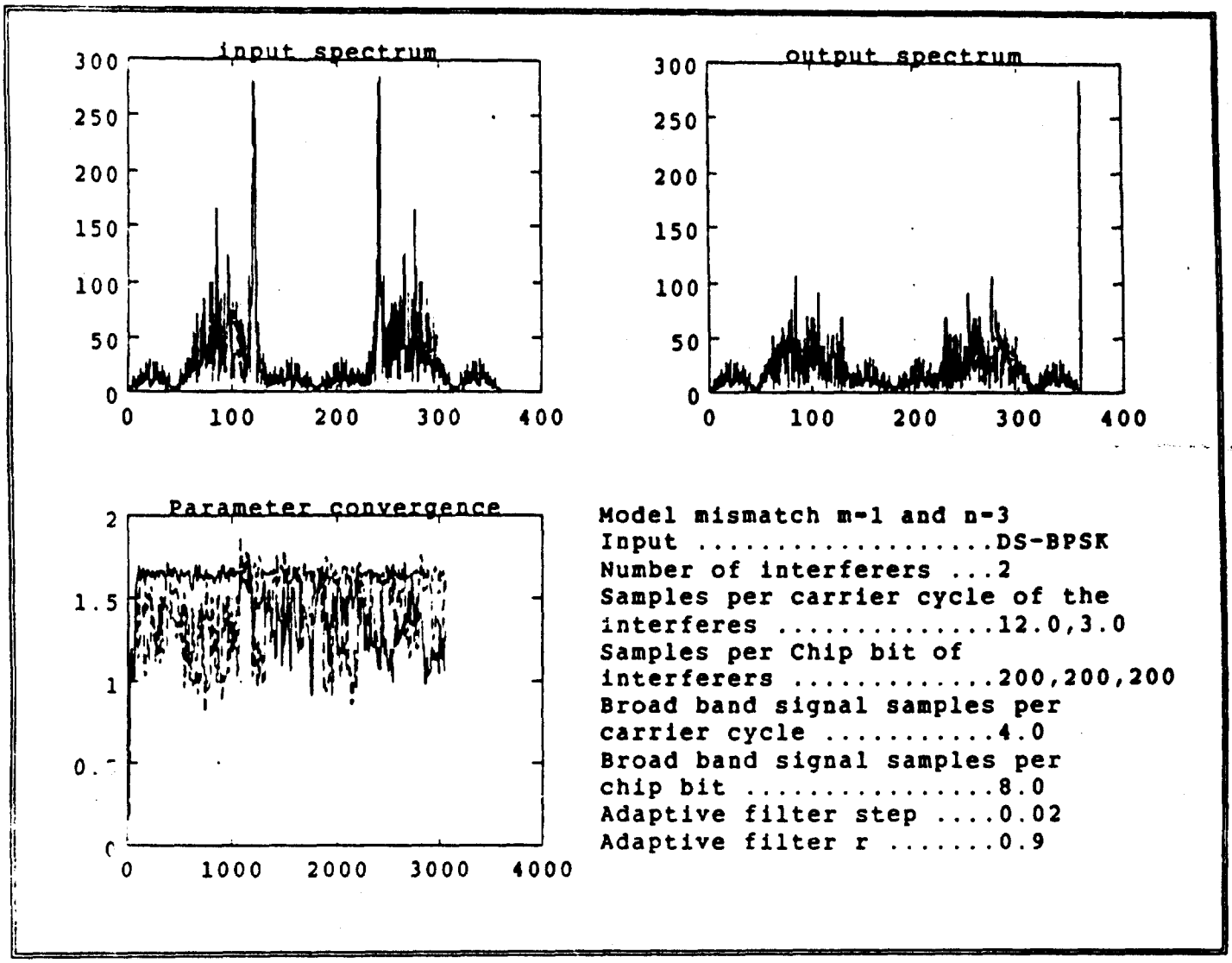


Figure 4-9. One Interferer and Three Notches.

5.0 Conclusions

In this report we have taken a careful look at the possible solutions to the problem of eliminating narrow-band interference in broad-band communications systems in the presence of white Gaussian noise. We have come to the following important conclusions:

1. Of the many possible algorithms available for solving this problem, only two seem to meet the requirements and also be feasible in terms of hardware complexity and hardware speed. These two approaches are:

   a. adaptive second-order notch filters based on a modified Kwan-Martin approach or

   b. an approach based on the DFT.

2. This report has studied carefully the adaptive notch filter approach based on a modification of Kwan and Martin and found it to be extremely well suited to the problem.

   a. Single interferers are eliminated within at most a few hundred iterations,

   b. With multiple interferers, the first notch will eliminate one of the interferers within a few hundred iterations and other notches kick in quickly such that as many as ten interferers can be eliminated in as few as 500 iterations,

   c. When there are fewer notches than interferers, notches can be designed using proper forgetting filters to either eliminate the maximum number of interferers or to attempt to reduce all interferers while eliminating some of them,

   d. When there are more notches than interferers, the extra notches do not adversely effect the broad-band signal,

e. Notches track well changing interferers and are able to handle frequency jumps that are less than one order of magnitude lower in frequency than the interferer signal frequency,

f. Single notches can eliminate two or more interferers, if those interferers are close together, and

g. The new algorithm is easily implemented on and IBM-PC or a VAX computer and seems well suited for hardware implementation.

5.1 Brief Look at Hardware Feasibility

Although hardware design is beyond the scope of this report, we shall look briefly at possible hardware configurations in order to satisfy ourselves that the algorithm is feasible for the desired application. Ideally, we would like to design a 5 to 10 notch filter that would operate at sampling frequencies up to 100 MHz. With currently available technology, about the best available commercially today would be 10 to 20 MHz. In what follows, we shall attempt to justify this 10-20 MHz figure and discuss how through the use of pipelining we could increase this sampling rate to as much as eight times the basic sampling rate (ie: 80-160MHz).

5.1.1 Commercially Available Hardware

In our investigation of the hardware feasibility of a 10-20 MHz sampling rate modified Kwan-Martin filter with 5 to 10 notch filters, we shall look at five commercially available DSP chips:

1. MIPS R3010, 40ns cycle, 2 cycle add, 5 cycle multiply, and 19 cycle divide [14],

2. Weitek 3364, 50ns cycle, 2 cycle add, 2 cycle multiply, and 17 cycle divide (30 cycle square root) [14],

3. Texas Instruments TI 8847, 30ns cycle, 2 cycle add, 3 cycle multiply, and 11 cycle divide (14 cycle square root) [14],

4. AT&T DSP32c, 20ns cycle, 2 cycle add, 2 cycle multiply, and 3 cycle divide [15], and

5. Honeywell HDSP 66110, 80ns cycle, performs vector dot product of 8 elements for an effective 10ns multiply and accumulate time [16].

The first four chips are of similar architecture and thus could be implemented using the same block diagram. The Honeywell chip is really an array processor, and would require us to vectorize the algorithm to take advantage of its speed. Although we believe that this vectorization could be done, it is beyond the scope of this report, so we shall concentrate our attention for the time being on the other four chips.

Figure 5-1 shows the block diagram for implementing a second-order filter using one of the first four DSP chips. The block diagram makes use of a coefficient memory which can be set by and external device. Multiplication A is done in parallel with Multiplication B and additions are done in sequence. The second-order building block of Figure 5-1 is used in Figure 5-2 to obtain one section of the modified Kwan-Martin Filter. This can then be repeated for all sections and results in a very regular structure as indicated in Figure 5-3. In the process of designing this, it was realized that an additional advantage of the modified algorithm over the original Kwan-Martin algorithm is that the original algorithm could not be broken down into identical blocks because of the need for the cascade chain in calculating the filter derivative [10].

5.1.2 Time Budget

The details of the time budget have been worked out in the M.S. Thesis [10]. The result is that the best floating point processing for the first four chips is achieved with the AT&T DSP-32c which yields a sampling rate of 5 MHz. This is not sufficient to meet our 10-20 MHz design requirement. However,

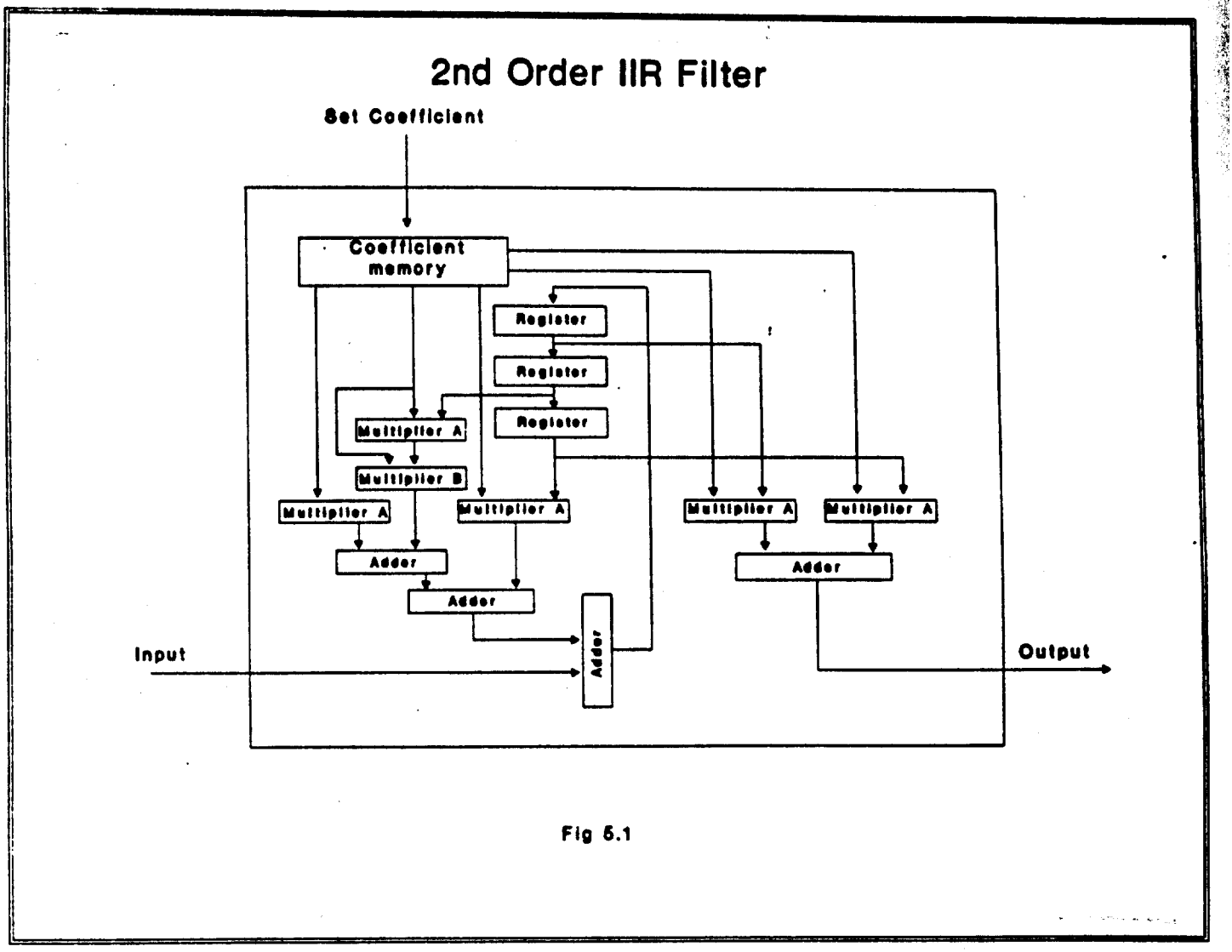


Fig 5.1

Figure 5-1. Block Diagram of 2nd Order IIR Filter

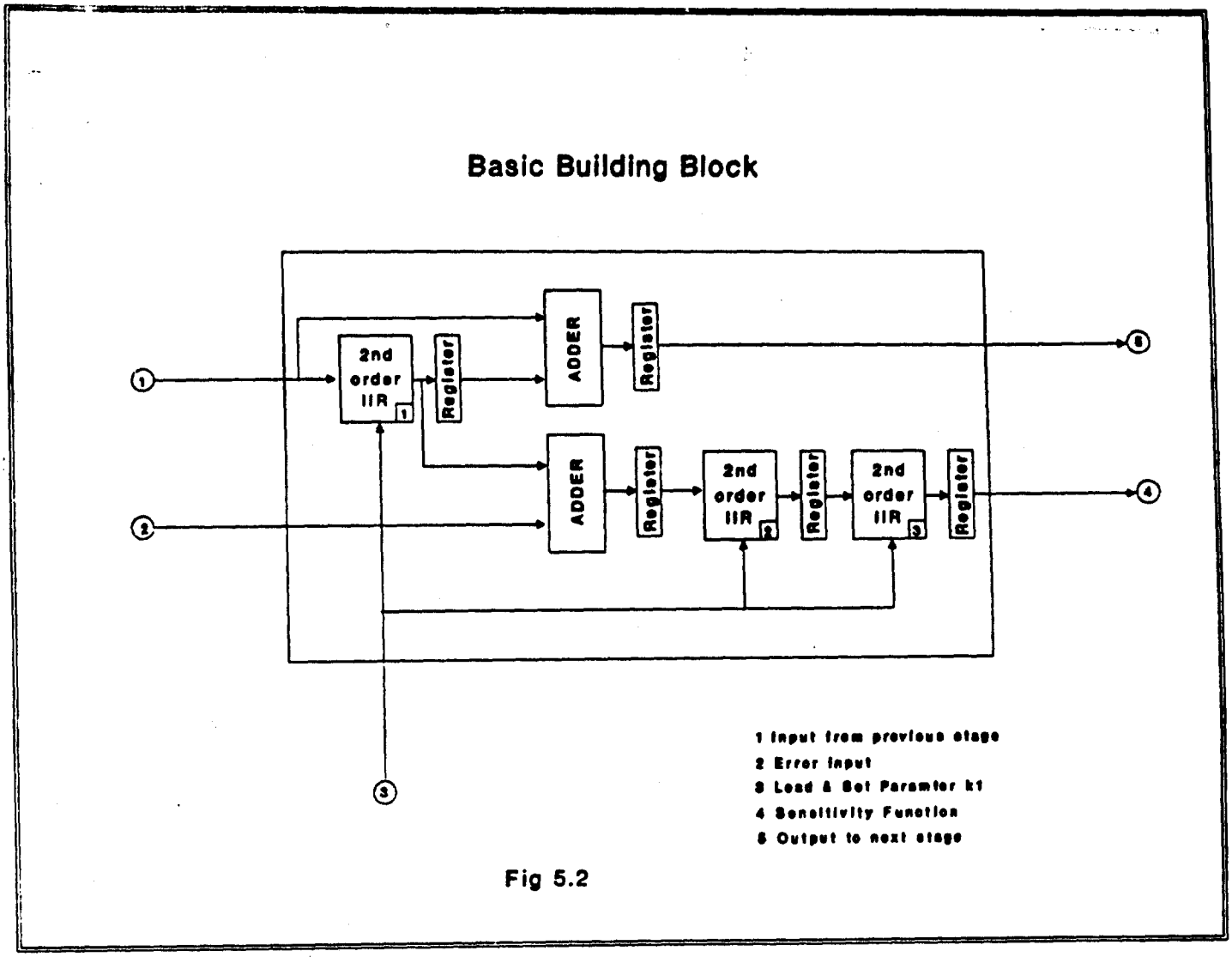


Fig 5.2

Figure 5-2. Block Diagram of Basic Building Block

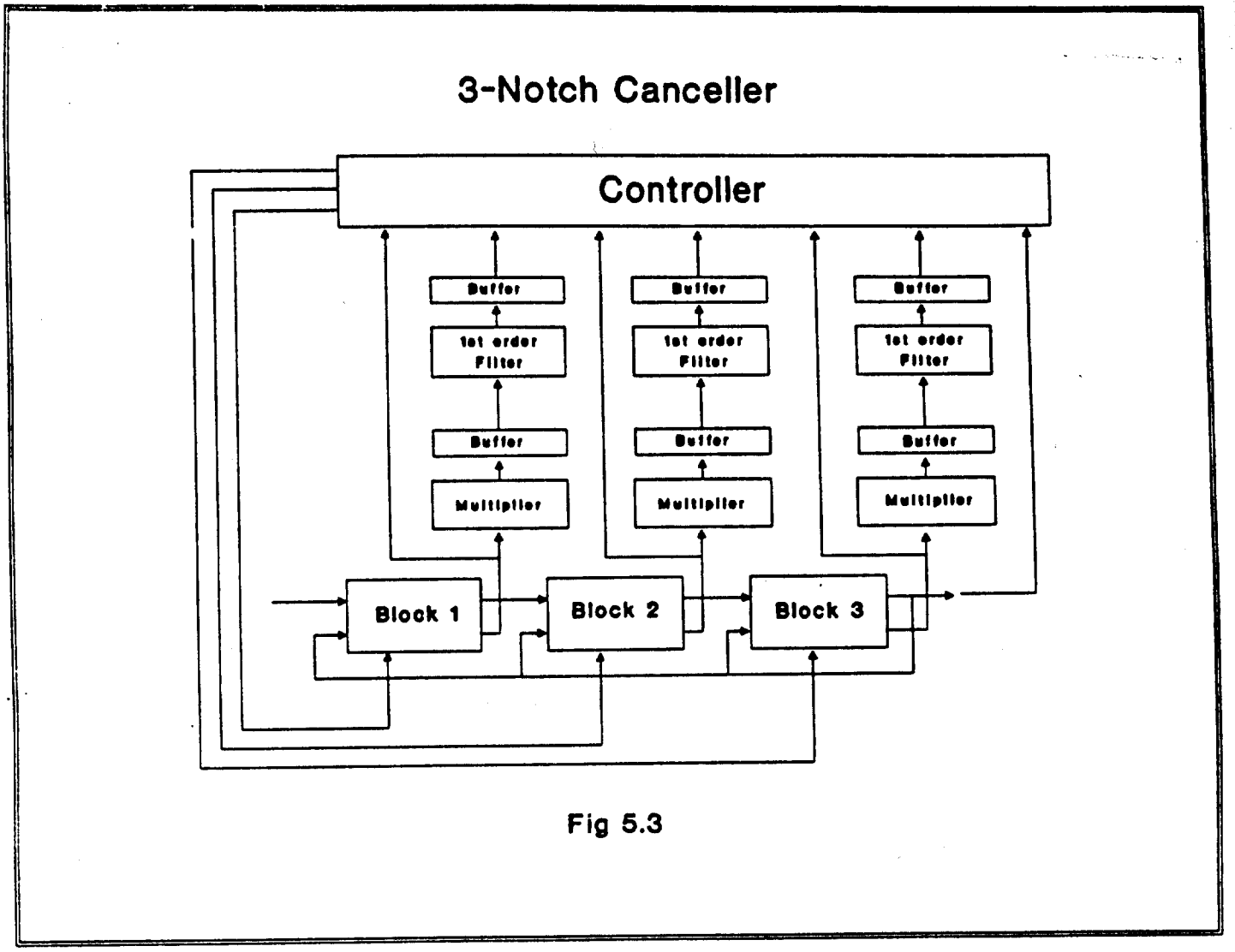


Fig 5.3

Figure 5-3. Block Diagram for 3-Notch Canceler

this was done without pipelining the IIR section. Using the pipelining techniques of Soderstrand and Loomis [17-21], we would multiply this sampling rate by at least two and probably four resulting in the desired 10-20 MHz design. Furthermore, if we were to vectorize the algorithm, the Honeywell chip could potentially yield 100 MHz sampling with both vectorization and pipelining. Based on this hardware analysis, we feel confident that sampling rates of 25 MHz or more can be achieved in practice with commercially available DSP chips or most certainly with a custom VLSI design.

5.2 Future Work

There are two avenues of future work that should be pursued before making a hardware commitment for the narrow-band filter project:

1. DFT based approaches to the problem should be studied, especially in view of the availability of high-speed FFT processors and such exotic approaches at recursive DFT's using QRNS arithmetic [9], and

2. a detailed investigation of hardware implementations of the modified Kwan-Martin algorithm should be made including simulation and layout of a custom VLSI approach. Because of the regularity of the structure in Figures 5-1 though 5-3, the custom VLSI may well yield some very excellent filters.

6.0 References

1. B. Friedlander and J.O. Smith, "Analysis and Performance Evaluation of an Adaptive Notch Filter," IEEE Trans. Information Theory, vol. IT-30, No. 2, pp. 283-295, March 1984.

2. D.V. Bhaskar Rao and S.Y. Kung, "Adaptive Notch for the Retrieval of Sinusoids in Noise," IEEE Trans. Acoustics Speech and Signal Proc., vol. ASSP-32, no. 4, pp. 791-802, August 1984.

3. A. Nehorai, "A Minimal Parameter Adaptive notch Filter With Constrained Poles and Zeros," IEEE Trans. Acoustics Speech and Signal Proc., vol. ASSP-33, no. 4, pp. 983-996, August 1985.

4. D.R. Hush, N. Ahmed, R. David and S.D. Stearns, "An Adaptive IIR Structure for Sinusoidal Enhancement, Frequency Estimation, and Detection," IEEE Trans. Acoustics Speech and Signal Proc., vol. ASSP-34, no. 6, pp. 1380-1390, December 1986.

5. T. Kwan and K.W. Martin, "Adaptive Detection and Enhancement of Multiple Sinusoids Using a Cascade IIR Filter," IEEE Trans. CAS, vol. CAS-36, no. 7, pp. 937-945, July 1989.

6. S. Nishimura, J.K. Kim and K. Hirano, "Mean-Squared Error Analysis of an Adaptive Notch Filter," Proc. 1989 IEEE Intern. Symp. CAS, Portland, OR, pp. 732-735, May 1989.

7. M.R. Petraglia, S.K. Mitra and J. Szczupak, "Adaptive Sinusoid Detection Using IIR Notch Filters and MultiRate Techniques," Proc. 1990 IEEE Intern. Symp. CAS, New Orleans, LA, pp. 271-274, May 1990.

8. M.R. Petraglia, J.J. Shynk and S.K. Mitra, "Stability Bounds for an Adaptive Notch Filter," Proc. 1990 IEEE Intern. Symp. CAS, New Orleans, LA, pp. 1963-1966, May 1990.

9. M.A. Soderstrand, W.K. Jenkins, G.A. Jullien, and F.J. Taylor, Residue Number Arithmetic and its Application to Modern Digital Signal Processing, IEEE Press, 1986.

10. K.V. Rangarao, Adaptive Digital Notch Filtering, M.S. Thesis, Naval Postgraduate School, Monterey, CA 93943, September 1991.

11. M.A. Soderstrand, H.H. Loomis, and K.V. Rangarao, "Improved Real-Time Adaptive Detection, Enhancement, or Elimination of Multiple Sinusoids", Proc. IEEE Midwest Symp. Circuits and Systems, Monterey, CA, May 1991.

12. M.A. Soderstrand, H.H. Loomis, and K.V. Rangarao, "Elimination of Narrow-Band Interference in BPSK-Modulated Signal Reception", Proc. IEEE Intern. Symp. Circuits and Systems, Singapore, June 1991.

13. Schell and Spooner, Cyclic Spectral Analysis Software Package Users Manual, Statistical Signal Proc., Inc, Yountville CA, June 30, 1990.

14. D.A. Patterson and J.L. Hennessy (with a contribution by David Goldberg), Computer Architecture: A Quantitative Approach, Morgan Kaufman Publishers, SSan Mateo, CA, 1990.

15. AT&T, Digital Signal Processor Information Manual for the WE DSP-32c, Morgan Kaufman Publication, Inc. San Mateo, CA 1990.

16. Honeywell, Technical Manual HDSP66110 and HDSP66210 Digital Signal Processing Chips, Honeywell, Inc, 1990.

17. B. Sinha and H.H. Loomis, "High-Speed Recursive Digital Filter Realizations," Circuits, Systems and Signal Processing, 1984.

18. M.A. Soderstrand and B. Sinha, "Pipelined Recursive RNS Digital Filter," IEEE Trans. CAS, vol. CAS-31, no. 4, pp. 415-417, April 1984.

19. K. Chopper and M.A. Soderstrand, "Implementation of Very High Speed Recursive Digital Filters," Proc. Asilomar Conf. on Circuits, Systems, and Computers, Nov. 1984.

20. K.K Parhi and D.G. Messerschmitt, "Pipeline Interleaving and Parallelism in Recursive Digital Filters," IEEE Trans. Acoustics Speech and Signal Proc., pp. 1099-1117, July 1989.

21. M.A. Soderstrand, H.H. Loomis and R. Gnanasekaran, "Pipelining Techniques for IIR Digital Filters," Proc. IEEE Intern. Symp. CAS, New Orleans, LA, pp. 121-124, May 1990.

Initial Distribution List

		Copies
1.	Secretary of the Air Force ATTN: Mr. Gary Peck, ACBMB The Pentagon, Room 5C132 Washington, DC 20330-1000	1
2.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
3.	Naval Postgraduate School ATTN: Dudley Knox Library, Code 52 Monterey, CA 93943	2
4.	Naval Postgraduate School ATTN: Research Office, Code 08 Monterey, CA 93943	1
5.	Naval Postgraduate School ATTN: Prof. Herschel Loomis Code EC/Lm Monterey, CA 93943	50
6.	Dr. Michael Soderstrand Department of Electrical Engineering and Computer Science University of California Davis, CA 95616	20